REVISTA PARA USUARIOS DE

Drean Commodore

AÑO 1 Nº 11 ₳ 2,70
REP. ARGENTINA




OCTUBRE 1986
Nº 19 ₳ 3,— REP. ARGENTINA
K64
COMPUTACION PARA TODOS
Mercado Argentino del Software: Aplicaciones Profesionales, Comerciales, Educativas y juegos para Commodore, CZ, TK, MSX, TI y Atari
K64 en Brasil
Arte y Ciencia
Suplemento Educativo
Radiografía de la C - 64
La Atari 130XE
Idiomas para Home Computers
¿Cómo son las IBM PC?

SUMARIO

NOTAS TECNICAS

PROGRAMAS

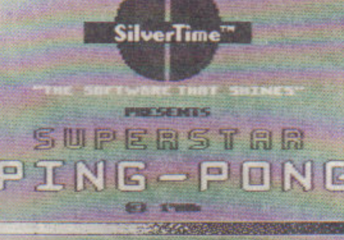
REVISION DE SOFT



Vamos a desentrañar los secretos de esta poderosa computadora y en particular cómo un chip (el UDM) organiza todo internamente para poder funcionar como si fuesen tres máquinas en una.

```
,FF00 DD 8D 07 CMP $078D,X
,FF03 DD 4C 59 CMP $594C,X
,FF06 EF       ???
,FF07 AD 95 02 LDA $0295
,FF0A 8D 06 DD STA $DD06
,FF0D AD 96 02 LDA $0296
,FF10 8D 07 DD STA $DD07
,FF13 A9 11    LDA #$11
,FF15 8D 0F DD STA $DD0F
,FF18 A9 12    LDA #$12
,FF1A 4D A1 02 EOR $02A1
,FF1D 8D A1 02 STA $02A1
,FF20 A9 FF    LDA #$FF
,FF22 8D 0
,FF25 8D 0
```

En diferentes notas les damos a los usuarios los elementos necesarios para programar con la 16, 64 y 128.





SECCIONES FIJAS

Ofrecemos las últimas novedades del mundo de los juegos. Todos los temas y escenarios se instalan en nuestras páginas. Podremos enterarnos desde las luchas de los caballeros del Rey Arturo hasta cómo comandar un submarino y, en el interín, jugar un partido al ping-pong.

Revista para usuarios de

Drean Commodore

AÑO 1 Nº 11 OCTUBRE DE 1986

Director General
Ernesto del Castillo

Director Editorial
Cristián Pusso

Director Periodístico
Fernando Flores

Director Financiero:
Javier Campos Malbrán

Arte y Diagramación
Fernando Amengual
Tamara Migelson

Secretario de Redacción
Ariel Testori

Redacción
Cristián Parodi

Fotografía
Víctor Grubicy
Eduardo Comesaña
Image Bank

Departamento de Avisos
Oscar Devoto
Nelso Capello

Departamento de Publicidad
Guillermo González Aldalur

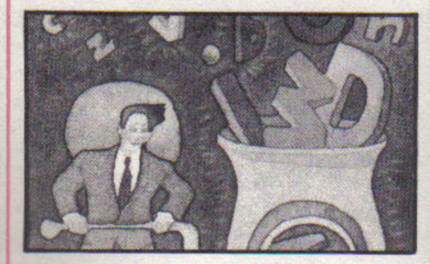
Revista para Usuarios de Drean Commodore es una publicación mensual editada por editorial **PROEDI S.A.**, Paraná 720, 5° Pis. (1017) Buenos Aires. Tel.: 46-2886 y 49-7130. Reg. Nac. de la Prop. Intelectual E.T., M. Registrada. Precio de este ejemplar, ₳ 2,70. Los ejemplares atrasados se venderán al precio del último número en circulación.

Impresión: Calcotam. Fotocromo tapa: Columbia. Fotocomposición: **Interamericana Gráfica SRL.**

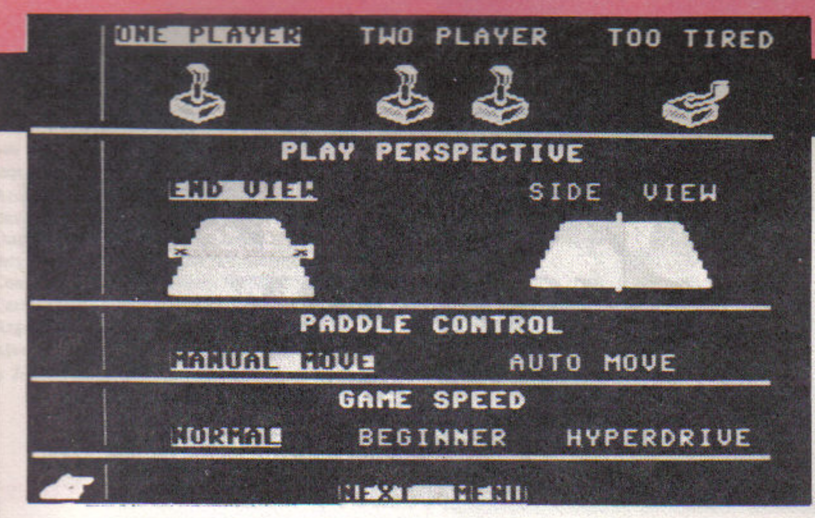
Distribuidor en Capital: Martino, Juan de Garay 358, P.B. Capital. Distribuidor interior: DGP. Hipólito Yrigoyen 1450, Capital Fededal. T.E. 38-9266/9800.

DE LA "A" A LA "Z"

Continuamos investigando sobre las variables string. Profundizando sobre el tema les explicamos cuál es el procedimiento para ordenarlas alfabéticamente.



En el número anterior hemos dado una introducción sobre qué son y cómo manejar las variables string o variables alfanúmericas.
Definimos a un string como una formación de números y letras, cualquiera sean éstas, y con una longitud no mayor a 256 caracteres.
La longitud se determina, simplemente, sumando la cantidad de caracteres. Así, por ejemplo, si hacemos:
A$="123456789"
la longitud de la variable es de 9 caracteres. Por suerte existe un comando Basic (conocido como LEN) que devuelve la longitud de un string o de una variable string.
Para el mismo ejemplo si hacemos en modo directo:
PRINT LEN(A$)
el sistema imprimirá 9.
En el número anterior también habíamos comentado cómo tomar una parte de una variable string a través de los comandos RIGTH$, LEFT$ y MID$.
Sin abandonar este importantísimo tema explicamos, en este mes, cuál es el procedimiento que se lleva a cabo cuando queremos ordenar alfabéticamente dos string.
Así como entre los números existe un orden natural (por ejemplo, 0 menor que 1, 1 menor que 2, etc) también existe ese orden con los caracteres alfanuméricos.
En las computadoras también se respeta el hecho de que A está antes de la B, B antes de la C y así sucesivamente.
Con esto queremos decir que se puede escribir sentencias del tipo:
IF A$ > B$ THEN.
Como cada carácter está asociado con un número dado por el ASCII (código normalizado americano para el intercambio de la información), el intérprete compara los caracteres como si fuesen números.
En realidad compara los códigos de esos caracteres, viendo si son iguales o distintos (mayor o menor).
En función del resultado de dicha comparación es que se determina el orden.
En otro orden de cosas explicamos a fondo como trabaja el UDM de la Commodore 128.
Como ustedes saben, la C-128 puede trabajar en tres modos diferentes: como una C-64 (totalmente compatible), como una C-128 y, finalmente, en modo CP/M.
Cuando la computadora funciona en cada uno de esos modos, necesita distintas configuraciones de memoria.
Para ello se ha diseñado la UDM (Unidad Directoria de memoria o, si la quieren en inglés MMU - Memory Manager Unit) que organiza las distintas configuraciones.
También continuamos explicando cómo acceder a las subrutinas internas de la Drean Commodore 64.
Gracias a cómo se han diseñado los equipos Drean Commodore, todas disponen de un sistema operativo (denominado Kernal) el cual es, igual para todas ellas.
Las subrutinas nos permiten ahorrar trabajo ya que usamos las ya hechas por Commodore.
Sólo necesitamos saber en qué dirección se encuentran y cómo pasar los parámetros que necesitan.
Recordemos que los parámetros son valores que se pasan desde el programa principal a la subrutina.

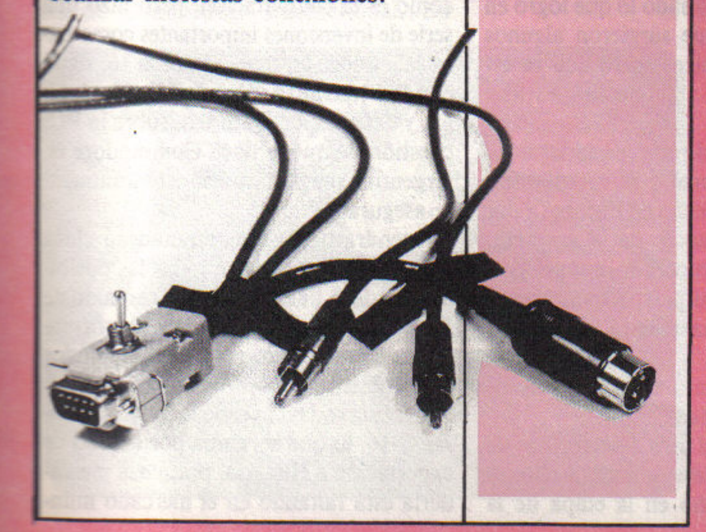


Ping Pong Super Profesional

Una vez más DATAGAMES SOFTWARE nos ha acercado los últimos juegos para la Drean Commodore 64. Entre ellos podemos destacar Star Ping Pong (comentado en la Sección Revisión de Software de este número).
El juego se destaca fundamentalmente por la calidad de gráficos y del movimiento de las paletas.
También nos ha llegado Time Trax, Chimera, Mission Elevator y Murder on the Mississippi.
Estos dos últimos también han sido comentados en la misma sección del número que tienen en sus manos.

Cable Conmutador 40-80 Columnas

La firma JDC Computación nos ha entregado una interface especial para poder utilizar la Commodore 128 con un monitor de 40-80 columnas.
Esta computadora trabaja en tres modos: el modo 64 sólo permite trabajar en 40 columnas mientras que los restantes dos pueden hacerlo en 80. Este cable tiene una llave que conmuta de 40 a 80 columnas.
De esta manera si estamos trabajando en modo 128 (80 columnas) y deseamos pasar a modo 64, sólo debemos mover una pequeña palanca en vez de realizar molestas conexiones.



Todos ellos caracterizados por su estupenda música e increíbles efectos gráficos.
Para mayor información llamar al 824-1060 u 821-5438 interno 31.

Basket

Seguramente se acordarán del famosísimo ONE ON ONE.
Aquel gran juego de basket desarrollado para la Commodore 64 y, desde ya, compatible con la Drean.
Sus creadores ofrecen una versión mejorada, el TWO ON TWO.
Según los que saben, éste supera ampliamente al anterior.
Como siempre, lo tendremos próximamente en Revisión de Software.
Para mayor información deberán llamar a Julio y Eduardo al 35-6106 o concurrir a la calle Corrientes 1148 2° J.

LA EXPANSION DE DREAN COMMODORE

Felipe Mc Gough

Les damos buenas noticias para los usuarios de estas máquinas, que incluyen desde el inminente lanzamiento de computadoras hasta la producción de software y un eficiente servicio técnico. También se proyecta la exportación de hardware. Informamos además sobre los entretelones de una reunión "top level" realizada en la turística Hunacau.

Este año que llega a su fin fue muy especial para Drean Commodore. Luego de un éxito arrollador, en el que las ventas de computadoras llegaron a cifras sobresalientes, no pocos fueron los rumores que rodearon este "boom". Que la producción no alcanzaba continuidad, que no existía un verdadero apoyo de la casa matriz y, aun más, que en Commodore International las cosas no iban nada bien, eran algunos de los "dardos envenenados" que surcaron el ambiente de las computadoras hogareñas. Para terminar con los trascendidos y dar una información nueva y exacta a los commodorianos es que entrevistamos a Felipe Mc Gough, Gerente de Publicidad y Promoción de Drean. Entre el 26 y 28 de septiembre pasado Mc Gough asistió a la reunión de distribuidores de Commodore para Latinoamericana. El cónclave se realizó en Hunacau, complejo hotelero a una hora de viaje de San Juan de Puerto Rico. Estaban representados Argentina, Chile, Costa Rica, Ecuador, Méjico, Perú, y España y Brasil como invitados especiales.

En relación al evento, Mc Gough expresó que "nunca se había hecho, porque nunca se había llegado a una etapa de organizarse, ya que justamente Latinoamérica está en la primera etapa de la computación". La idea de unificar criterios en forma integradora es positiva al existir casos como el de Ecuador, que todavía no recibió su primera partida de computadoras.

El encuentro surgió a instancias de Tom Rattigan, presidente de Commodore y también de Nick Bessy, responsable en Inglaterra pero abarcando una gran área de Commodore en Europa y sobretodo en Latinoamérica.

El ritmo de trabajo fue intenso y se desató luego de la bienvenida brindada por Joe Alves, un portugués que trabaja en los Estados Unidos para Commodore, en Westchester, Philadelphia, responsable de toda el área latinoamericana.

LAS BUENAS NOTICIAS VIENEN DE PUERTO RICO

Tom Rattigan fue quien dio a conoceer la situación de Commodore International con motivo de todo lo que se había comentado. Commodore había crecido desde 1983 hasta 1985 y en el año fiscal '86, que para Estados Unidos terminó en junio, había tenido una gran pérdida que le costó la cabeza a algunos altos directivos de la empresa, sobre todo a algunos responsables de Latinoamérica, que ya dejaron la compañía.

Para Rattigan lo importante, entonces, era saber cuál era la situación en los países latinoamericanos, qué pretendían, qué habían recibido, y asegurarles que Commodore no había ganado lo que logró en años anteriores, que surgieron algunos problemas económicos, pero que ya estaban resueltos y todo estaba solucionándose.

Además, estaba ahí para respaldar a todos los distribuidores y representantes, cosa que ya había hecho en Europa, y dar a conocer los balances de la empresa, cuánto se había recuperado, cuánto perdido, cuáles eran las perspectivas para el año '87, y el lanzamiento de nuevos modelos.

La reunión se orientó a ver la situación en los distintos países.

En computación, el más desarrollado de los presentes era Méjico, aunque ellos se han quedado mucho en la etapa de la C-16, que ha tenido un éxito fantástico, por lo que siguen trabajándola (aunque están muy avanzados en la C-128, la Amiga y la PC).

EL TURNO DE ARGENTINA

En cuanto a volúmenes, a pesar de todo, Argentina que todavía no tiene doce meses de representación de Commodore, es la que mayor cantidad alcanzó haciendo la proporción entre tiempo y número de computadoras vendidas. Esto provocó un interés muy grande por parte de la gente de Commodore, sobre todo teniendo en cuenta la diferencia de población entre Méjico y nuestro país.

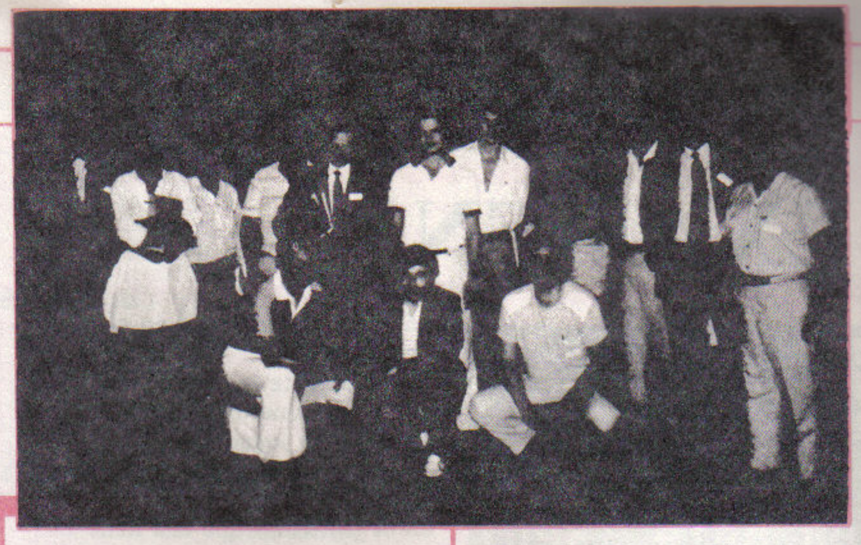
Nosotros superamos, a pesar de la ventaja azteca en tiempo, ampliamente las pretensiones que ellos tenían puestas en Argentina. Se interesaron mucho en todo lo que Drean desarrolló, principalmente ahora para la línea nueva.

Mc Gough dice al respecto que "Drean está en el primer año de un área nueva como es la computación, pero hizo una serie de inversiones importantes como para salir adelante, empezó por la 16, siguó con la 64, avanza sobre la 128 y la Amiga, y después, en el futuro, sobre la PC, cuestión de que la línea Commodore en Argentina para el usuario esté totalmente asegurada".

Así tendrá desde la computadora chica hasta la PC grande con todas las posibilidades de la línea y aparte el respaldo de Drean, que ya tiene el apoyo confirmado de Commodore para eliminar todos los problemas de service, piratería, etc.

Argentina todavía vende las computadoras C-16, lo que abre una posibilidad de exportación a Hungría, porque la mercadería está faltando en el mercado mun-

LLEGA LA AMIGA



Los asistentes a la reunión de Puerto Rico. De izquierda a derecha: Ruben Meiller (Eduador), Armando Morell y Miguel Luna (Costa Rica), Nick Bessy (Commodore International), Alejandro Guzmán (Costa Rica), Felipe Mc Gough (Drean/ Argentina), Tom Rattigan (Presidente de Commodore International), Roberto Baesa (Chile), Carlos Domenech (España), Carlos Barros (Drean/ Argentina). Abajo: Alfredo Acuña (Méjico), Joe Alves (Commodore U.S.A. para América Latina) y Jorge Pando (Perú).

dial por el avance de la 64.
Siguiendo en el plano de las negociaciones, países como Costa Rica, que son chicos y traen todo de Estados Unidos, están interesados por el gran desarrollo que se hizo en Argentina y Méjico en cuanto a manuales, software, todo en español, porque todo lo que reciben va en inglés. A raíz de esta reunión surgió la posibilidad de realizar una exportación a Chile tanto de hardware como de software y lo mismo podría suceder con Ecuador y Perú.
El encuentro sirvió para confirmar que la Amiga es el producto sobre el que la Commodore ha volcado mayor interés y donde tiene puestas sus expectativas. "Indudablemente las prestaciones, las posibilidades y la gama de la máquina son espectaculares —afirmó Mc Gough— por lo que Drean no se mantiene ajena a ese hecho".
De acuerdo a lo trascendido, en muy corto plazo la Amiga va a estar oficialmente en la Argentina con todo el apoyo de Drean y Commodore.
La empresa está en las últimas negociaciones para tenerla en el país posiblemente a fines de este año. Cuando Drean presente oficialmente la continuidad de la línea en Argentina va a haber representantes de Commodore International. En marzo vendrá Nick Bessy, y posiblemente en agosto Tom Rattigan visite las instalaciones de Drean y su planta de San Luis.

EL FUTURO

La afinidad y el entendimiento puesto de manifiesto en Puerto Rico dejó sentadas las bases para un nuevo cónclave de la gente de Commodore a nivel latinoamericano. La próxima reunión será en julio de 1987 en Méjico. En tanto se trabaja para que exista una unidad entre todas las partes.



Commodore Amiga

"Estamos en una etapa de reorganización —dijo Mc Gough—, de terminar la última etapa de la 16 y 64, estudiar, ver que posibilidades de exportación hay, y volcarnos a otras áreas como educación".
En relación al software, Drean cree que primero hay que regularizarlo y entonces sí encarar su producción. Está en conversaciones con importantes compañías mundiales para tomar sus derechos y que de esa manera el usuario de Commodore tenga el soft Drean Commodore oficial.
La empresa garantizará respaldo, continuidad, service y también la eliminación de los piratas. Una vez hecho eso el proyecto es comprar los derechos para toda Latinoamérica, cuestión de poder comercializarlo en todos los países.
En cuanto al service, las dudas llegarán a su fin. El directivo expresó que "el usuario de Jujuy va a tener su service en Jujuy que le garantice, ante cualquier duda que tenga, dónde recurrir, porque de nada sirve que compre una máquina en Buenos Aires y después no la pueda reparar en su provincia".

EXAMEN

```
7000 DATA COMAN
JUEGOS DE CARTA
7010 DATA CUERP
ES IMPORTANTES,
7020 DATA ISLAS
7500 DATA INPUT
SS,OPENS,CMDS,G
7510 DATA PIPA,
RNELN
7600 DATA HUMER
PARIETALS,FRONT
7610 DATA CLAVI
7620 DATA CRANE
N,HIPOFISISN
7700 DATA MUSS,
UTES,CANASTAS,E
7710 DATA LA BR
7720 DATA LA PI
LLON,RULETAN,EL
7800 DATA PINOS
ACIAS,SECUOYAS,
```

Comp: Drean Commodore 64
Conf: Basica
Tipo: Educativo

Para aquellos que quieran probar sus conocimientos generales, aquí les presentamos un programa educativo que les permitirá practicar aún más. Pueden participar uno, dos o tres jugadores. A cada uno de ellos se los interrogará sobre si ciertas palabras pertenecen o no al tema expuesto.

Primeramente la computadora nos pedirá que ingresemos la cantidad de jugadores que participarán (1-3).

Luego ingresará a su base de datos los temas que ella tiene almacenados en sentencias DATA.

A continuación pondrá un tema e irá imprimiendo palabras. Nosotros, a través de las teclas N y S, deberemos indicarle a la computadora si esa palabra pertenece o no al tema en cuestión.

Si acertamos, el borde de la pantalla se pondrá verde, posibilitándonos continuar dentro del juego.

Si nos equivocamos, se pondrá roja. En este caso se imprimirá el puntaje obtenido.

Variables utilizadas

Nombre	Descripción
NC	Tecla oprimida
T,I	Variable auxiliar
TM,P.	Contadores
P	Puntaje obtenido
C	Indice
G$	Detecta tecla
A$,B$	Matrices de trabajo
D$	Vector auxiliar

Descripción del programa

Línea	Descripción
100-130	Prepara pantalla y DATAS a leer
350-430	Pregunta cantidad de jugadores y verifica que el valor ingresado sea 1, 2 ó 3.
500	Retardo
850-900	Lee DATAS
1000-1950	Rutinas varias
2000-2540	Rutina: Lee los datos almacenados y los pone en las matrices respectivas
2600-2999	Imprime resultado y sigue con el siguiente jugador
3000-3410	Comienza a preguntar e imprimir palabras
3430-3460	Verifica teclas S y N
7000-8420	Base de datos. Palabras a interrogar

```
1 REM ************************
2 REM *        EXAMEN         *
3 REM ************************
100 CLR:PRINT"■■"
120 POKE53280,6:POKE53281,6
130 RESTORE
350 PRINT"■■■■■■■■■■■■        CUANTOS JUGADORES ? (1-3)■";
400 GETG$:IFG$=""THEN400
420 NC=ASC(G$)-48
430 IF(NC<1)OR(NC>3)THEN400
440 PRINTNC
500 FORT=0TO1000:NEXT
850 PRINT"■■■■■■■■■■■■  LEYENDO DATOS. UN MOMENTO..."
900 GOTO2000
1000 PRINT"                                       □"
1020 PRINTTAB(20-LA/2)A$"□"
1030 RETURN
1900 FORT=0TO4000:NEXT
1930 H=1:GOSUB5700
1950 GOTO3300
2000 REM
2020 DIMA$(15,10),B$(15,10),D$(10)
2050 FORPR=1TO10
```

```
2080 READD$(PR)
2100 NEXT
2120 FORPR=1TO10
2140 FORRE=1TO15
2160 READA$(RE,PR)
2180 B$(RE,PR)=RIGHT$(A$(RE,PR),1)
2200 NEXT
2220 NEXT
2301 C$(1)="PRIMER"
2302 C$(2)="SEGUNDO"
2303 C$(3)="TERCER"
2350 C2=C2+1:C=C2-INT((C2-1)/NC)*NC
2400 PRINT"                    ";C$(C);" JUGADOR"
2450 PRINT"                          PREPARADO"
2460 PRINT""
2470 PRINT"              OPRIMA UNA TECLA "
2490 POKE198,0
2500 GETG$:IFG$=""THEN2500
2540 GOTO3000
2600 PRINT"                  ";C$(C);" JUGADOR"
2620 P(C)=P(C)+P
2630 PRINT"             PUNTOS";P(C),"    RONDA";INT((C2-1)/NC)+1
2640 IFC2=3*NCTHEN2700
2645 IFNC=1THEN2690
2650 PRINT"                   SIGUIENTE JUGADOR"
2690 FORT=0TO3000:NEXT:GOTO2350
2700 FORT=0TO3000:NEXT:GOTO4000
2999 END
3000 PRINT""
3020 POKE53280,15
3150 PR=INT(10*RND(0))+1
3160 IFD$(PR)="X"THEN3150
3180 A$=D$(PR):D$(PR)="X"
3200 LA=LEN(A$)
3220 GOSUB1000
3250 FORT=0TO4000:NEXT
3260 PRINT"":A$="S       N":LA=8:GOSUB1000
3280 PRINT""
3290 TM=0:P=0
3300 RE=INT(15*RND(0)+1)
3320 IFA$(RE,PR)="X"THEN3300
3340 LA=LEN(A$(RE,PR))-1
3360 A$=MID$(A$(RE,PR),1,LA)
3370 A$(RE,PR)="X":Q=0
3380 GOSUB1000
3390 POKE198,0
3400 POKE1829+TM,81
3410 TM=TM+1:IFTM=29THEN3900
3430 GETG$
3450 IFG$="S"THEN3600
3460 IFG$="N"THEN3700
3470 IFPEEK(56320)AND16THEN3472
3471 GOTO3600
3472 IFPEEK(56321)AND16THEN3480
3473 GOTO3600
3480 IF(PEEK(56320)AND15)<>15THEN3700
3481 IF(PEEK(56321)AND15)<>15THEN3700
3490 Q=Q+1:IFQ=8THENQ=0:GOTO3400
3500 GOTO3430
3600 IFB$(RE,PR)="S"THEN3800
3680 GOTO3900
3700 IFB$(RE,PR)="N"THEN3800
3780 GOTO3900
```

```
3800 POKE53280,5
3820 FORT=0TO500:NEXT
3885 TM=TM-1:P=P+25:IFP>350THEN4500
3890 POKE53280,14:GOTO3300
3900 POKE53280,2
3910 FORT=0TO1000:NEXT
3920 GOTO2600
3999 REM
4000 PRINT"                    PUNTUACIONES"
4010 PRINT"                   ____________"
4030 FORC=1TONC
4040 PRINT"          ";C$(C)" JUGADOR",P(C)
4060 NEXT
4110 PRINT"              OPRIMA UNA TECLA "
4130 GETG$:IFG$=""THEN4130
4200 PRINT"                   OTRA PARTIDA ?"
4230 GETG$:IFG$=""THEN4230
4250 IFG$="S"THEN100
4270 IFG$="N"THENEND
4499 REM
4500 POKE53280,14
4520 A$="ENHORABUENA"
4530 LA=LEN(A$)
4550 GOSUB1000
4850 FORT=0TO1000:NEXT
4900 GOTO2600
5500 A$=C$(C)+" CONCURSANTE"
5520 LA=LEN(A$)
5540 GOSUB1000
5550 PRINT""
5690 RETURN
7000 DATA COMANDOS BASIC,HUESOS HUMANOS,JUEGOS DE CARTAS,ARBOLES
7010 DATA CUERPOS DEL SISTEMA SOLAR,MONTES IMPORTANTES,PAISES,RIOS
7020 DATA ISLAS,ESTRECHOS Y CANALES
7500 DATA INPUTS,PRINTS,GET$S,RIGHT$S,SYSS,OPENS,CMDS,GOTOS,READS,STATUSS
7510 DATA PIPN,FLIP FLOPN,JUMPN,FINDN,KERNELN
7600 DATA HUMEROS,TIBIAS,FEMURS,PERONES,PARIETALS,FRONTALS,HOMOPLATOS,ROTULAS
7610 DATA CLAVICULAS,CALCANEOS
7620 DATA CRANEON,MENISCON,GEMELOSN,CODON,HIPOFISISN
7700 DATA MUSS,POKERS,KINGS,LA ESCOBAS,TUTES,CANASTAS,EL BURROS,BRIDGES
7710 DATA LA BRISCAS,BLACK JACKS
7720 DATA LA PITOTAN,EL PALEN,EL CARAMBILLON,RULETAN,EL PUNTO Y RAYAN
7800 DATA PINOS,CHOPOS,ABETOS,CIPRESS,ACACIAS,SECUOYAS,OLIVOS,SAUCES
7810 DATA ALCORNOQUES,NOGALS
7820 DATA FRESALN,HELECHON,JARAN,BREVERAN,CALANDRION
7900 DATA VENUSS,MARTES,MERCURIOS,JUPITERS,SATURNOS,URANOS,TITANS,EUROPAS
7910 DATA LA LUNAS,PLUTONS,CERESS
7920 DATA OSA MAYORN,SIRION,VIA LACTEAN,APOLLON
8000 DATA MCKINLEYS,MONT BLANCS,TEIDES,MULHACENS,ANETOS,ETNAS,VELETAS,EVERESTS
8010 DATA K2S,KILIMANJAROS,ACONCAGUAS
8020 DATA BORNEON,BELICEN,SIERRA LEONAN,CALIFORNIAN
8100 DATA URSSS,CANADAS,CHINAS,EE.UU.S,BRASILS,BIRMANIAS,TAILANDIAS,ECUADORS
8110 DATA NEPALS,COSTA DE MARFILS,LIBERIAS
8120 DATA AMERICA DEL NORTEN,OCEANIAN,ANTARTIDAN,BENELUXN
8200 DATA NILOS,AMAZONASS,MISSISSIPPIS,AMARILLOS,RIO DE LA PLATAS,PARANAS,AMURS
8210 DATA MACKENZIES,LENAS,NIGERS,MISSOURIS
8220 DATA MUERTON,RIO DEL ORON,SARMIENTON,SALINASN
8300 DATA GROENLANDIAS,NUEVA GUINEAS,BORNEOS,MADAGASCARS,SUMATRAS,CELEBESS
8310 DATA TASMANIAS,SAJALINS,DEVONS,TIERRA DEL FUEGOS,CANARIASS
8320 DATA MONTECARLON,DINAMARCAN,YOSEMITEN,VOSTOKN
8400 DATA DAVISS,HUDSONS,YUCATANS,MAGALLANESS,BERINGS,MALACAS,GIBRALTARS
8410 DATA MANCHAS,PASO DRAKES,LUZONS,BASSS,MOZAMBIQUES
8420 DATA MIGERN,ORINOCON,ALASKAN
```

BUSCADO

Dream Commodore busca al usuario del año '86



CAUSA

Premiaremos a las aplicaciones más originales de las Commodore 16 y 64 en cualquier ámbito o actividad. Por ejemplo, se tendrán en cuenta el uso inteligente de estas máquinas en el campo profesional, educativo, comercial, científico y doméstico.

Quienes deseen participar deberán enviar a nuestra redacción, Paraná 720, 5to. Piso, Cap. Federal (1017), una descripción del uso que se le dio al equipo (si es posible con fotos).

El cierre del certamen será el 28/11/86

RECOMPENSA

1er. PREMIO: UNA COMPUTADORA DREAN COMMODORE 64

2do. PREMIO: UNA DISKETTERA DREAN COMMODORE

COMO ORDENAR ALFABETICAMENTE

Ya vimos que las variables string son conjuntos concatenados de letras y números con un propósito específico. Ahora les explicamos cómo, utilizando esta herramienta, podemos ordenar alfabéticamente.

El orden alfabético debe ser, seguramente, lo único que nos acompaña durante toda nuestra existencia. Al ingresar a la escuela primaria, éste nos obliga a esperar para decir "presente" cuando se nos toma lista.

En la escuela secundaria, nos sigue obligando como antes pero, cosa que antes no aportaba, nos da una mano y es la de poder especular cuando, por orden de lista, se nos toma lección.

Dichosos los que, como Zapetini, gozan de esa "inmunidad" implícita porque su apellido comienza con una Z (sólo estudian cuando se están "acercando"). No sucede lo mismo con los Abate, siempre los primeros.

Si continuamos analizando las famosas listas, seguramente ocuparíamos todo el espacio disponible para esta sección.

En esta nota explicaremos temas que involucran al orden alfabético antes mencionado. Por supuesto, y sin escaparnos mucho de nuestra temática específica, orientado a cómo es en la computación. Más específicamente, cómo es en los equipos Drean Commodore.

En el número anterior ya habíamos dado una introducción explicando qué son los string.

Los definimos como un conjunto de letras y números; en otras palabras una formación alfanumérica, cuya longitud no superara los 256 caracteres.

Explicamos cómo concatenarlos y cómo manejar a gusto subexpresiones provenientes de ellos, es decir



cómo "cortarlos" y tomar partes de ellos.

Una de las cosas que no mencionamos es si es posible determinar qué string es alfabéticamente anterior a otro.

Antes de iniciar la explicación de este tema debemos hacer una pausa y comentar qué es el código ASCII y cómo son las comparaciones "normales".

Al decir comparaciones normales nos estamos refiriendo a las comparaciones del tipo A>B, A = B o A>B, siendo A y B dos variables numericas comunes.

Para los que desconozcan los anteriores símbolos, ellos denotan cuándo un número es mayor, igual o menor respectivamente.

Así, por ejemplo, 5 es mayor que 3 (5>3), igual a sí mismo (5=5) y menor que 8 (5<8).

En lo que respecta al código ASCII (siglas de American Standard Code for Information Interchange -código americano normalizado para el intercambio de información), representa números, letras y caracteres especiales.

Así, a cada caracter se lo representa a través de un número, cuidando que éste mantenga la "proporción" con el alfabeto.

Con esto queremos decir que si la A está antes de la B, entonces el código que se le asigna a la A debe ser menor que el que se le asigna a la B. Y es así como sucede. El código ASCII de la A es 65 y el de la B es 66, 67 para la C, 68 para D y así sucesivamente.

Con los números sucede lo mismo, con la salvedad de que los códigos dados son menores que los de las letras, pero siempre respetando la condición de "menor" que existe entre ellos (0<1<2<3<4<5<6<7<8<9). Así el código ASCII del 0 es 48, el del 1 es 49, el del 2 es 50 así hasta llegar al 9 cuyo ASCII es 57.

De acuerdo con esto, y retomando nuestro tema principal, el de las comparaciones, al intérprete le resulta relativamente fácil determinar cuál es alfabéticamente anterior.

Analicemos el siguiente caso, y veamos cómo trabaja el intérprete Basic cuando hace una comparación. Tenemos dos string. Por un lado la variable A$ cuyo contenido es "AAAB" y B$ cuyo contenido es "AAAC".

Cuando a través de la sentencia IF-

THEN preguntamos si IFA$<B$THEN... el intérprete hace lo siguiente:
1) Toma el primer caracter de la variable A$ (en nuestro caso A) y lo "enfrenta" con el primer caracter de B$ (también A).
2) Compara los códigos ASCII de cada uno de ellos. Si el código del caracter de A$ es menor o mayor que el de B$, entonces finaliza la comparación, pudiendo resultar que A$ es alfabéticamente anterior o posterior a B$.
3) Si ambos caracteres son iguales (como en nuestro caso) se procede a tomar los siguientes caracteres de las variables.
4) Si alguna de ellas ya no tiene más caracteres, se determina a la que se quedó sin caracteres como la alfabéticamente menor.
5) Si todavía hay más caracteres se repite la operación desde el punto 2.
Continuando en nuestro caso, resulta pues que A$ es menor (alfabéticamente) que B$.
Ambas variables tienen los tres primeros caracteres iguales. Recién el cuarto es distinto siendo su ASCII menor que el de B$.
Una prueba rápida para confirmar este resultado es hacer en modo directo:
PRINT "AAAB'>"AAAC".
Los operadores relacionales como el "<' y el ">" devuelven un valor de – 1 si la proposición es verdadera o 0 si es falsa.
En el caso anterior el intérprete devolverá -1.
Una posible aplicación consiste en la calsificación de datos alfanuméricos.
Imaginemos tener un vector en donde se encuentran 100 nombres totalmente desordenados. Nosotros deberemos hacer un programa que los ordene alfabéticamente.
El programa que realiza esto último corresponde al listado 1.
Suponemos que los nombres ya se encuentran dentro del vector, al cual denominamos NOM$. Este también se encuentra correctamente dimensionado.
Así como trabajamos sobre un vector podemos hacerlo sobre un archivo en disco o en cassette.
En próximos números continuaremos abordando temas de interés para neófitos y programadores expertos.

Listado 1

```
10 FORI=1TO99

20 FORJ=I+1TO100

30 IFNOM$(J)<NOM$(I)
THENAUX$=NOM$(J):NOM$
(J)=NOM$(I):NOM$(I)=AUX$

30 NEXTJ

40 NEXTI

50 REM VECTOR ORDENADO
```

EL MONITOR ASSEMBLER

Al igual que la C-128, la 16 dispone de un monitor interno que le permite al programador trabajar en Assembler. En esta nota les comentamos qué es y para qué sirve.

```
AC002  A2  FF      LDX  #$FF
AC004  20  D2  FF  JSR  $FFD2
AC007  A9  23      LDA  #$23
AC009  20  D2  FF  JSR  $FFD2
AC00C  24  23      BIT  $23
AC00E  D0  F0      BNE  $C000
AC010  90  1C      BCC  $C02E
AC012  18          CLC
AC013  20  AA  EF  JSR  $EFAA
```

Seguramente casi todos los usuarios de la Drean Commodore 16 se habrán preguntado qué ese "monitor" que se nombra en el manual de la computadora.
Algunos, los más osados, tipean MONITOR y observan detalladamente lo que sucede.
Ven cómo aparece una serie de palabras con números y letras, y signo de interrogación (?) cuando comenzamos a tipear cosas que, obviamente, no entiende el monitor.
Luego de imprimirse varios "?" en la pantalla, deciden abandonar el proceso ya sea a través de una suave presión sobre el botón del reset o de un violentísimo corte de suministro eléctrico.
Los de mayor suerte, pueden volver al Basic debido a que tipearon el comando que permite tal tarea.
Y aquí paramos, por ahora. No queremos crear confusiones. Es decir, no queremos que los lectores confundan monitor "para ver" con el monitor que aquí explicamos.
El primero es un periférico de salida, es decir es el que nos permite ver lo que tipeamos.

El otro, el que nos interesa a nosotros, nos permite escribir programas de otra manera. Esa manera es usando el Assembler como lenguaje de programación.

Listado

```
A 3000 LDX #$00

A 3002 LDA $4000,X

A 3005 CMP #$00

A 3007 BEQ $3010

A 3009 JSR $FFD2

A 300C INX

A 300D JMP $3002

A 3010 RTS


> 4000 48 4F 4C 41

00 00 00 00
```

UN POCO DE HISTORIA CON ASSEMBLER

La persona que quería programar hace unos 40 años atrás debía ser, primeramente, programador, y tener bastante conocimientos de electrónica.
No olvidemos que una computadora estaba formada por varios miles de lámparas, las cuales "emanaban" bastante calor.
Ese calor colaboraba a disminuir la vida útil de cada una de ellas. Por ello a cada rato debían suspender el procesamiento y cambiar la lámpara quemada.
Asi como existían limitaciones de hardware también había limitaciones de software.
Primeramente los programas se ingresaban manipulando una serie de llaves. Después, con la aparición del microprocesador, se comenzó a trabajar en Código Máquina (lo único que comprende la computadora).
Un día alguien se dio cuenta de que era más fácil ingresar un cierto código que estar ingresando ceros y unos. Asi se fijó un objetivo: cuando se ingresara un código (mucho más fácil de manejar) se debería traducir a su equivalente en código máquina.
De esta manera es, más o menos, como surge el Assembler. Es decir un lenguaje que acepta un código bien definido y devuelve su correspondiente en código máquina.
A ese código se le puso "mnemónico". Se trató que tuviese sentido con respecto a las instrucciones que representaba.
Es decir que si el objetivo de una instrucción consiste en cargar una variable con un dato, el mnemónico debe ser de tal manera que el programador entienda de qué se trata con sólo verlo. Para ese caso pudo ser LDA (del inglés Load —cargar—).
No hubiese tenido sentido poner un super código, que para entenderlo hay que pasar primero por un traductor (tipo diccionario) que nos dice qué es.

Por ejemplo, cargar una variable se representa a través de 00AX-QW1/082. ¿Se imagina cuál sería el costo de comprender los listados?
Hasta acá comprendimos qué es el Assembler y cómo podemos en la Drean Commodore 16 usar el monitor para escribir programas en dicho lenguaje.
Una acotación al margen, no todas las home computers le permiten al usuario escribir programas en Assembler.
Una de las principales características del lenguaje es que como cada mnemónico se traduce a su correspondiente en código máquina, el programa se ejecuta más rápido ya que la computadora no pierde tiempo en traducciones, como las que se efectúan en Basic.
Claro que también están los puntos en contra. Con un ejemplo lo veremos más claro. El listado 1 es un programa en Assembler que imprime la palabra HOLA en pantalla.
Observen todo lo que se necesita para hacer esta simple tarea.
Desde el Basic esto lo hacemos a través de:
PRINT "HOLA"
Es decir que el Assembler es más veloz y no es fácil escribir programas utilizándolo.
Para hacer nuestra tarea un poco más llevadera, la Drean Commodore 16 nos suministra un programa traductor, llamado comunmente monitor.
Este nos permite escribir programas en Assembler y, luego de ingresar cada instrucción, traducirla a su correspondiente código de máquina.
Así como el Basic dispone de LIST, NEW, PRINT, etc., el monitor tiene sus propios comandos. A cada uno de ellos se los representa a través de una sola letra.
Por ejemplo, para ver una sección de memoria se utiliza el comando M. Podemos indicar también desde dónde a dónde queremos examinar. O si queremos visualizar los registros internos del microprocesador, deberemos usar el comando R.
Así como podemos ensamblar, es decir pasar de un mnemónico a su código máquina, también podemos efectuar el pasaje inverso; de código máquina a mnemónico. Este procedimiento se llama desensamblar.
Como el LOAD y SAVE, el monitor tiene sus comandos L y S. Aquí la diferencia radica en que con los segundos podemos grabar o cargar determinadas secciones de memoria.
Finalmente debemos decir que para poder utilizar el monitor debemos saber, primero, Assembler. Un tema tratado por nuestra publicación en anteriores números.
Por ahora ya sabemos algo de este misterioso lenguaje, que no olvidaremos y seguiremos tratando próximamente.

LISTADOR

```
   2 REM CUIDADO QUE SE AUTOBORRA
   3 REM GRABARLO ANTES DE EJECUTARLO
   4 REM O CAMBIAR LINEA 210 POR STOP
 100 DEV = 4 :
     REM NUMERO DE DISPOSITIVO DE LA IM
PRESORA
 110 LOC = 49152 :
     LIN = 1000
 120 PRINT "POKEANDO DATOS DE LA LINEA"
;LIN
 125 K = 0 :
     SUM = 0
 130 IF LOC > 50148 THEN 170
 140 READ N :
     POKE LOC,N
 145 LOC = LOC + 1 :
     SUM = SUM + N :
     K = K + 1
 148 IF K < 16 THEN 130
 150 READ N :
```

Comp : DC 64
Tipo : Utilitario
Conf : Basica

A medida que nuestros programas van aumentando en tamaño, se hace necesario insertar espacios y comentarios que ayuden, un poco, a clarificarlos.
El programa que aquí les presentamos es un utilitario que ayuda al programador a comprender sus listados.
Lo que hace es insertar espacios entre las sentencias, imprimir una sola sentencia por línea en caso de que se hayan puesto varias y, lo más útil, separa los lazos FOR-NEXT.
Para que esto se entienda mejor visualicemos los listados 2 y 3.
En el primero, los lazos FOR-NEXT están insertados uno dentro del anterior con lo que determinar la correspondencia de cada NEXT con su respectivo FOR se hace un tanto delicada. Más aún si entre cada una de éstas existe una serie de sentencias, como en este caso.
El listado 3 representa cómo se verá el listado 2 si se utiliza el LISTADOR. Noten cómo se comprende mejor la función del programa.

Explicación del programa

Línea	Descripción
1-4	Comentarios del programa
100-110	Inicializa variables a utilizar
120	Indica que línea se esta POKEando
125-170	Comprueba que se esten ingresando en memoria valores correctos
190-200	Indica que se ha cargado el listador
210	Borra el programa
1000-1620	Instrucción del programa escritas en código máquina

Variables utilizadas

Nombre	Descripción
DEV	Dispositivo impresora
LOC	Dirección de inicio de carga de instrucciones
LIN	Línea inicial de las instrucciones en DATAs
SUM	Sumatoria de los valores leídos
N	Ultimo valor leído por en cada DATA

El listado 1 corresponde al LISTADOR. Carguen y grábenlo antes de ejecutarlo ya que se autoborra una vez que se POKEa en memoria.
Para activarlo sólo se debe tipear SYS49152. El programa preguntará si deseamos sacar el listado por printer (impresora) o por screen (pantalla).
Si optamos por la primera opción, se nos interrogará por el título que tendrá cada página. Así, cada una tendrá ese título junto con su correspondiente número.
Para ambos casos el listado se puede suspender oprimiendo cualquier tecla. Para brequearlo solo se debe oprimir la tecla 'Q'.

Listado 2

```
10 FOR I=1 TO 10
20 FOR J=1 TO 10
30 PRINT I,J
40 FOR K=1 TO 10
50 PRINT K
60 NEXT
70 NEXT
80 NEXT
```

Listado 3

```
10 FOR I=1 TO 10
20  FOR J=1 TO 10
30   PRINT I,J
40    FOR K=1 TO 10
50     PRINT K
60    NEXT
70  NEXT
80 NEXT
```

Listado 1

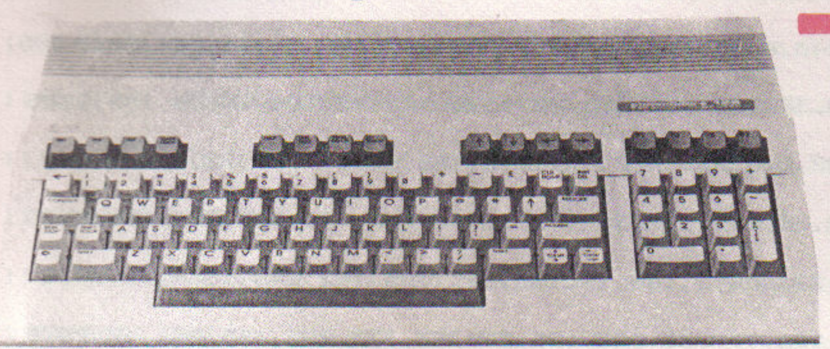
```
1 REM *** LISTADOR ***
2 REM CUIDADO QUE SE AUTOBORRA
3 REM GRABARLO ANTES DE EJECUTARLO
4 REM O CAMBIAR LINEA 210 POR STOP
100 DEV=4:REM NUMERO DE DISPOSITIVO DE LA IMPRESORA
110 LOC=49152:LIN=1000
120 PRINT"POKEANDO DATOS DE LA LINEA";LIN
125 K=0:SUM=0
130 IFLOC>50148THEN170
140 READN:POKELOC,N
145 LOC=LOC+1:SUM=SUM+N:K=K+1
148 IFK<16THEN130
150 READN:IFN=SUMTHENLIN=LIN+10:GOTO120
160 PRINT"ERROR EN EL DATA DE LA LINEA";LIN:PRINT"VERIFIQUE LOS VALORES":STO
170 READN:IFN<>SUMTHEN160
180 POKE49250,DEV
190 PRINT"!cr ab!PROGRAMA !rvs on!LISTADOR!rvs off! CARGADO"
200 PRINT" TIPEE !rvs on!SYS49152!rvs off! PARA ACTIVARLO"
210 NEW
1000 DATA 162,0,189,140,195,240,6,32,210,255,232,208,245,32,228,255,2629
```

```
1010 DATA 240,251,201,83,208,11,141,232,195,169,147,32,201,194,76,120,2501
1020 DATA 192,201,80,208,232,141,232,195,32,210,255,32,199,194,162,0,2565
1030 DATA 189,177,195,240,6,32,210,255,232,208,245,162,61,169,0,157,2538
1040 DATA 254,195,202,169,32,157,254,195,202,208,250,32,207,255,201,13,2826
1050 DATA 240,8,157,254,195,232,224,60,208,241,169,1,141,244,195,169,2738
1060 DATA 3,162,4,160,255,32,186,255,169,0,32,189,255,32,192,255,2181
1070 DATA 162,3,32,201,255,32,91,194,165,43,133,251,165,44,133,252,2156
1080 DATA 198,251,169,0,141,233,195,169,6,141,236,195,173,233,195,208,2743
1090 DATA 39,160,0,169,0,141,230,195,141,245,195,141,234,195,141,246,2472
1100 DATA 195,141,229,195,162,3,177,251,208,3,238,230,195,200,202,208,2837
1110 DATA 245,173,230,195,201,3,208,7,32,199,194,32,231,255,96,177,2478
1120 DATA 251,141,242,195,200,177,251,140,253,195,141,243,195,162,0,32,2818
1130 DATA 67,194,32,195,194,169,2,141,246,195,172,253,195,200,177,251,2683
1140 DATA 208,23,140,253,195,32,74,195,173,253,195,24,101,251,133,251,2501
1150 DATA 169,0,101,252,133,252,76,140,192,170,173,245,195,208,43,138,2487
1160 DATA 201,177,144,25,201,180,176,21,41,127,141,251,195,173,239,195,2487
1170 DATA 208,3,32,195,194,169,1,141,239,195,76,139,193,173,239,195,2392
1180 DATA 240,3,32,195,194,169,0,141,239,195,138,201,58,208,23,173,2209
1190 DATA 245,195,240,4,169,58,208,43,32,195,194,169,58,32,201,194,2237
1200 DATA 32,74,195,76,221,192,138,201,34,208,13,173,245,195,73,255,2325
1210 DATA 141,245,195,169,34,76,232,193,201,32,208,13,173,245,195,240,2592
1220 DATA 0,169,32,32,201,194,76,221,192,201,128,176,3,76,232,193,2126
1230 DATA 173,245,195,208,115,138,41,127,141,251,195,201,2,208,9,173,2422
1240 DATA 236,195,56,233,2,141,236,195,32,195,194,162,0,140,253,195,2465
1250 DATA 160,0,236,251,195,240,14,185,158,160,201,129,144,4,232,200,2509
1260 DATA 208,240,200,208,242,185,158,160,201,129,144,54,41,127,32,201,2530
1270 DATA 194,173,239,195,208,22,173,251,195,201,35,240,15,201,38,240,2620
1280 DATA 11,201,52,144,4,201,75,144,3,32,195,194,173,251,195,201,2076
1290 DATA 1,208,9,173,236,195,24,105,2,141,236,195,172,253,195,76,2221
1300 DATA 221,192,32,201,194,200,208,189,32,201,194,76,221,192,169,48,2570
1310 DATA 141,231,195,173,242,195,56,237,240,195,141,242,195,173,243,195,3094
```

```
1320 DATA 237,241,195,141,243,195,144,5,238,231,195,208,230,24,173,240,2940
1330 DATA 195,109,242,195,141,242,195,173,241,195,109,243,195,141,243,195,3054
1340 DATA 173,231,195,201,48,208,9,173,234,195,208,7,169,32,208,6,2237
1350 DATA 141,234,195,173,231,195,32,201,194,169,0,141,237,195,174,252,2764
1360 DATA 195,232,232,189,217,195,201,42,208,1,96,142,252,195,141,240,2778
1370 DATA 195,232,189,217,195,141,241,195,76,238,193,173,254,195,201,64,2999
1380 DATA 240,96,32,199,194,162,0,189,133,195,240,6,32,210,255,232,2415
1390 DATA 208,245,162,0,189,254,195,240,6,32,210,255,232,208,245,162,2843
1400 DATA 0,189,211,195,240,6,32,210,255,232,208,245,169,0,174,244,2610
1410 DATA 195,140,253,195,32,205,189,172,253,195,32,199,194,162,0,189,2605
1420 DATA 133,195,240,6,32,210,255,232,208,245,162,67,169,45,32,210,2441
1430 DATA 255,202,208,248,32,199,194,32,199,194,169,5,141,238,195,238,2749
1440 DATA 244,195,96,169,32,208,2,169,13,140,247,195,142,248,195,141,2436
1450 DATA 235,195,201,32,208,17,205,237,195,240,67,173,237,195,201,40,2678
1460 DATA 240,60,173,246,195,208,55,173,246,195,240,38,174,236,195,201,2875
1470 DATA 2,240,4,138,76,251,194,138,56,233,6,141,230,195,141,229,2274
1480 DATA 195,240,12,169,32,32,210,255,206,230,195,208,246,169,0,141,2540
1490 DATA 246,195,173,235,195,141,237,195,32,210,255,238,229,195,32,228,3036
1500 DATA 255,208,2,240,30,169,128,141,249,195,141,250,195,206,249,195,2853
1510 DATA 208,251,206,250,195,208,246,32,228,255,240,251,201,81,208,3,3063
1520 DATA 141,233,195,172,247,195,174,248,195,96,32,199,194,173,232,195,2921
1530 DATA 201,80,208,43,173,229,195,201,80,144,8,238,238,195,169,0,2402
1540 DATA 141,229,195,238,238,195,173,238,195,201,62,144,18,173,254,195,2889
1550 DATA 201,64,240,11,162,5,32,199,194,202,208,250,32,91,194,169,2254
1560 DATA 1,141,246,195,96,32,32,32,32,32,32,0,147,83,65,76,1242
1570 DATA 73,68,65,32,69,78,32,80,82,73,78,84,69,82,32,79,1076
1580 DATA 32,32,83,67,82,69,69,78,32,40,80,47,83,41,63,32,930
1590 DATA 0,73,78,71,82,69,83,69,32,69,76,32,84,73,84,85,1060
1600 DATA 76,79,32,68,69,32,80,65,71,73,78,65,32,32,32,32,916
1610 DATA 58,13,0,80,65,71,46,32,0,16,39,232,3,100,0,10,765
1620 DATA 0,1,0,42,42,85
```

EL REFEREE DE LA COMMODORE 128

Vamos a desentrañar los secretos de esta poderosa computadora y, en particular, cómo un chip (el UDM) organiza todo internamente para poder funcionar como si fuesen tres máquinas en una.



Sin lugar a dudas la casa Commodore se enfrentó a un compromiso realmente serio cuando decidió poner en el mercado una nueva máquina.

Más aún teniendo en cuenta el éxito obtenido por su predecesora —la C-64— de la que ya hay en el mercado norteamericano más de un millón de unidades. Con un toque de picardía, bastante habilidad comercial y mucho ingenio técnico se logró una nueva máquina. La C-128 dispone de novedades tales como el notable lenguaje Basic 7.0, capacidad de memoria RAM de 128 Kbytes y salida para monitor de 80 columnas.

Para capitalizar el suceso de la 64 incluyeron, además, una modalidad tipo 64 totalmente compatible con aquella máquina, lo que permite que los programas escritos para una puedan funcionar en la nueva.

Este hecho habrá entusiasmado a más de uno dado que podía disponer de una computadora superior sin desprenderse de su vieja máquina y sus programas. Como si esto fuera poco existe una tercera modalidad en la que se pueden ejecutar programas en CP/M versión 3.0 Plus.

Resumiendo: en una habitan 3 computadoras completas e independientes. Teniendo ya este panorama general vamos a introducirnos en la intimidad de esta maravilla.

Un poco de arquitectura

Si tuviéramos la posibilidad de echarle una mirada a la plaqueta que se encuentra en el interior de la computadora, descubriríamos la existencia de unos sesenta circuitos integrados o chips.

Algunos de éstos se encargan de realizar tareas específicas como generar los efectos de sonido, las señales de video, comunicarse con el exterior o almacenar información.

Otros se ocupan de conducir esa Información, y finalmente uno —el microprocesador— es el que se encarga de dirigir todo.

La commodore 128 cuenta con dos microprocesadores: el 8502 y el Z-80. Cada uno de ellos cumple una tarea específica. El 8502 —evolución del viejo 6502 que utiliza la 64— se encarga de sostener los modos de funcionamiento C-64 y C-128, mientras que el Z-80 supervisa la operación en el modo CP/M.

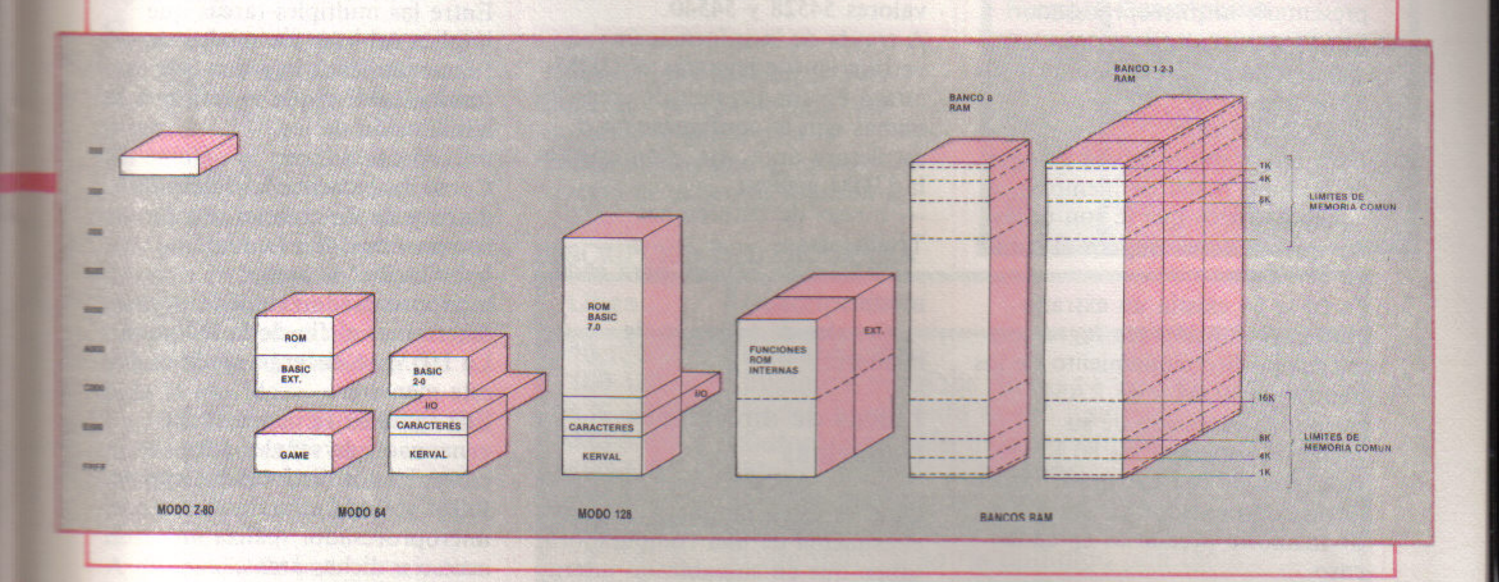
En cuanto a los sistemas operativos, es decir el conjunto de instrucciones que debe ejecutar el microprocesador cuando la máquina se enciende y que permite su posterior y correcto funcionamiento, vienen provistos de fábrica, como parte de la memoria ROM para los modos C-128 y C-64. En tanto que para que la máquina funcione en modo CP/M se le debe suministrar desde el exterior un programa adecuado.

Pero la parte fundamental de nuestra historia se halla relacionada con el tratamiento que se le da al resto de los chips (los mencionados generadores de video y sonido, interfaces y memorias)

Es fácil entender que razones de espacio y de costo impiden dedicarle a cada modo de funcionamiento una ''batería'' completa de dispositivos. Esta es la principal razón que determina que sólo un modo por vez pueda ser activado y tener acceso a tales medios operativos. Y al mismo tiempo es el motivo que obliga a elaborar un mecanismo interno que sea capaz de poner a disposición del microprocesador que corresponda tal o cual elemento necesario.

En otras palabras, toda esta ingeniosa maniobra de contener a tres computadoras en una, está

dada por un no menos ingenioso —y hasta complicado— manejo del campo de memoria.
Quizás alguno se pregunte dónde está la dificultad de ese manejo. Bien, tanto el microprocesador 8502 como el Z-80 tienen la posibilidad de seleccionar una posición de memoria entre 65535 (que son los famosos 64 Kbyte). Esta selección se realiza a través de 16 líneas, llamadas líneas de dirección. Como cada línea admite sólo dos estados posibles (0 a 1 —los famosos bits—), el total de combinaciones distintas es 2 16=65535. Precisamente cada combinación corresponde a una dirección particular que puede ser solicitada.
Solamente considerando los 128 Kbyte de memoria (algo así como 131072 bytes) excedemos holgadamente la capacidad de



direccionamiento de nuestros microprocesadores (recuerden que sólo pueden direccionar hasta 65535).
Frente a esto no nos queda más remedio que admitir que hay "gato encerrado".
Tiene que haber algún mecanismo que se encargue de presentarle al microprocesador los dispositivos que necesita en cada momento ajustándolos a sus posibilidades de direccionamiento.
Para este trabajo deberá escoger del extenso mapa de memoria, a aquellos trozos que le son útiles sin que su suma supere el límite de 64 Kbyte.
Ya en la 64 existía un extraño chip cuya función era lograr ese adecuado direccionamiento de los bancos de ROM y de RAM, porque la extensión de su memoria superaba los 80 Kbyte.
Era tarea de la PLA presentarle al microprocesador las secciones de memoria necesarias en cada caso.
En la 128 nos encontramos con otro nuevo y enigmático chip: la Unidad Directora de Memoria (UDM).
Su trabajo es nada menos que realizar la elección de las distintas configuraciones de memoria para el modo 128, lograr la mencionada compatibilidad 128/64 y administrar los medios operativos para cada modo, entre otras cosas.

La UDM por dentro

Como hemos anticipado, esta unidad realiza el mismo tipo de tarea que la PLA. Pero a diferencia de ésta, admite modificaciones a través de sus 12 registros que permiten obtener como resultado final el mapa de memoria necesario.
Llamamos registro a todo elemento que contiene información y puede ser consultada con facilidad. La información en este caso está relacionada con las maniobras que se realizarán en el campo de memoria.
Cada registro posee una longitud de ocho bits, donde cada bit puede tener un significado especial.
Se puede llegar a estos registros a través del sistema de direccionamiento, dado que ocupan una posición específica del mapa de memoria, entre los valores 54528 y 54540.
A través de la información vertida en los registros la UDM estará en condiciones de crear los mapas con la configuración o estructura necesaria. Con este fin la UDM genera:
— Líneas de Direcciones Traducidas
— Señales de control para cada modo
— Señales de selección de memoria

Líneas de direcciones traducidas

Quienes estén familiarizados con el esquema de una computadora saben que en su cableado interno pueden distinguirse 3 tipos de líneas. Las ya mencionadas líneas de dirección que seleccionan una posición entre otras, las líneas de datos que transportan la información entre distintos puntos del sistema y finalmente un conjunto de líneas llamadas de control que se encargan justamente de controlar y dirigir ciertas operaciones (por ejemplo, habilitar algún chip).
Pero en nuestro caso ese conjunto de líneas requiere un tratamiento adicional por parte de la UDM ya que deben adaptarse al modo de funcionamiento y al trabajo que se esté ejecutando.
Habíamos comprobado que la memoria total superaba a la capacidad de direccionamiento. Por ese motivo, la UDM actúa como un intermediario entre el microprocesador y el dispositivo deseado.
Cuando el microprocesador hace un llamado a un dispositivo mediante sus 16 líneas de dirección, la UDM lo intercepta.
Según sea la configuración de memoria que se esté utilizando, se encarga de ubicar en su verdadera dirección al dispositivo que fue solicitado por el micro para lo cual emplea las líneas de direcciones traducidas.

Señales de selección y control

Entre las múltiples tareas que implica arbitrar los modos de funcionamiento hay una que es fundamental y que se refiere a la habilitación de los microprocesadores.
Como físicamente las líneas de datos y de direcciones son compartidas, la eventual habilitación de ambos microprocesadores puede determinar el fin de la máquina.
La UDM se encarga también de este menester.
Ella también se ocupa de la generación de señales para controlar los distintos bancos de ROM y RAM cuando el microprocesador realiza un acceso a dichas áreas.
Para culminar, les dejamos el gráfico correspondiente a la figura 1. En ella se representa la organización del mapa de memoria de la C-128.

Pequeño glosario

Es nuestro deseo poner en sus manos la explicación de algunas palabras técnicas utilizadas en esta nota.
ROM: Memoria exclusiva para lectura. La provee el fabricante y la información que ella tiene almacenada no se puede modificar. RAM: Memoria que permite la escritura y la lectura. En ella se instalan los datos cuando se programa.
PLA: Permite "armar" distintas configuraciones de memoria.
Direccionamiento: Es la tarea que realiza el microprocesador cuando desea acceder a una determinada posición de memoria para luego tomar o poner información. Lo hace en base a un conjunto de líneas, llamadas líneas de direcciones.

Guillermo Fornaresio

TATETI

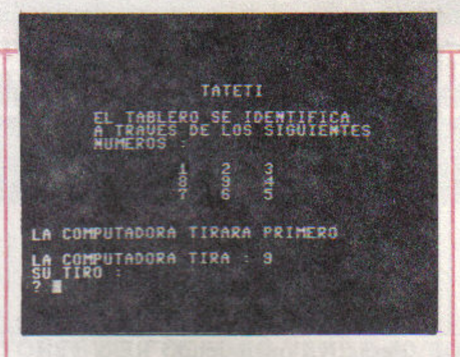
Comp: Drean Commodore 16
Conf: Básica
Tipo: Juego

Seguramente no hará falta explicar con lujo de detalle cuál es el objetivo de este tradicional juego.

En cambio es necesario decir cómo pide la computadora la entrada de nuestro movimiento. Ella enumera el tablero como se muestra en la figura 1. Por ejemplo, si queremos mover en el centro, deberemos contestar con 9 cuando la computadora nos pida la movida.

Fue diseñado para que juege un oponente humano versus la computadora. Aleatoriamente puede mover cualquiera de los dos.

A diferencia de otros tateti, éste fue desarrollado desde un aspecto heurístico del problema. Vale decir que no sólo evitará que nosotros le ganemos a la computadora sino que, además, tratará de ganarnos a nosotros.



TABLERO DE TATETI

1	2	3
8	9	4
7	6	5

VARIABLES UTILIZADAS

Nombre	Descripción
P,I,S,J	Punteros varios
N	Contador
C,B	Vectores juego TATETI
F	Flag. Indica quién movió
M	Indica jugada

DESCRIPCION DEL PROGRAMA

Línea	Descripción
10-130	Presentación
150-420	Inicializa matrices de juego
430-470	Reestablece para nuevo juego
480-550	Determina quién mueve primero
560-770	Mueve primero el jugador
778-920	Mueve primero la computadora
930-1200	Determina adónde mover
1210-1220	Determina si se ha oprimido una tecla

```
10 REM ********************
20 REM *       TATETI       *
30 REM ********************
40 PRINT"♥"
50 PRINT"■■■■"TAB(15)" TATETI
   ":PRINT
60 PRINT"        EL TABLERO SE
   IDENTIFICA"
70 PRINT"        A TRAVES DE LOS
   SIGUIENTES"
80 PRINT"        NUMEROS :"
90 PRINT
100 PRINTTAB(05);"           1   2   3"
110 PRINTTAB(05);"           8   9   4"
120 PRINTTAB(05);"           7   6   5"
130 PRINT:PRINT
140 RESTORE
150 REM INICIALIZAR MATRICES T Y U
160 FORP=1TO8
170 FORI=1TO3
180 READT(P,I)
190 NEXTI
200 NEXTP
210 DATA1,2,3
220 DATA8,9,4
230 DATA7,6,5
240 DATA1,8,7
250 DATA2,9,6
260 DATA3,4,5
270 DATA1,9,5
280 DATA7,9,3
290 FORS=1TO9
300 FORJ=1TO4
310 READU(S,J)
320 NEXTJ
330 NEXTS
340 DATA1,4,7,0
350 DATA1,5,0,0
360 DATA1,6,8,0
370 DATA2,6,0,0
380 DATA3,6,7,0
390 DATA3,5,0,0
400 DATA3,4,8,0
410 DATA2,4,0,0
420 DATA2,5,7,8
430 REM NUEVO JUEGO
440 N=0
450 FORS=1TO9
460 C(S)=0
470 B(S)=0:NEXTS
480 REM QUIEN EMPIEZA ?
490 IFRND(1)<.5THEN530
500 PRINT"USTED TIRARA PRIMERO"
510 PRINT
520 GOTO560
530 PRINT"LA COMPUTADORA TIRARA
    PRIMERO"
540 PRINT
550 GOTO780
560 REM TIRA EL JUGADOR
570 PRINT"SU TIRO :"
580 INPUTM
590 F=-1
600 IFM=INT(M)THEN630
610 PRINT"TIRO ILEGAL; INTENTE
    OTRA VEZ"
620 GOTO560
630 IFM<1THEN610
640 IFM>9THEN610
650 IFB(M)<>0THEN610
660 REM ACTUALIZA LISTA
670 B(M)=F
680 FORJ=1TO4
690 P=U(M,J)
700 IFP=0THEN740
710 C(P)=C(P)+F
720 IFC(P)=-3THEN890
730 IFC(P)=3THEN830
740 NEXTJ
750 N=N+1
760 IFN=9THEN910
770 IFF=1THEN560
780 REM TIRO DE LA COMPU
790 GOSUB930
800 PRINT"LA COMPUTADORA TIRA :";M
810 F=1
820 GOTO660
830 REM JUEGO ACABOSE
840 PRINT"Y LA COMPUTADORA GANA"
850 PRINT
860 PRINT
870 PRINT"NUEVO JUEGO":PRINT"OPRIME
    UNA TECLA":GuSUB1210:GOTO430
880 GOTO430
890 PRINT"ME GANASTE. CASUALIDAD..."
900 GOTO870
910 PRINT"JUEGO EMPATADO"
920 GOTO870
930 REM SELECCIONAR MOVIMIENTO
940 FORP=1TO8
950 IFC(P)=2THEN1010
960 NEXTP
970 FORP=1TO8
980 IFC(P)=-2THEN1010
990 NEXTP
1000 GOTO1050
1010 FORI=1TO3
1020 M=T(P,I)
1030 IFB(M)=0THEN1200
1040 NEXTI
1050 FORS=1TO9
1060 V(S)=0
1070 IFB(S)<>0THEN1130
1080 FORJ=1TO4
1090 P=U(S,J)
1100 IFP=0THEN1120
1110 V(S)=V(S)+ABS(C(P))+1
1120 NEXTJ
1130 NEXTS
1140 V=0
1150 FORS=1TO9
1160 IFV(S)<=VTHEN1190
1170 V=V(S)
1180 M=S
1190 NEXTS
1200 RETURN
1210 GETA$:IFA$=""THEN1210
1220 RETURN
READY.
```

TRUCOS

Modo comillas

Todos coincidirán en que cuando se oprimen comillas en los equipos Drean Commodore se entra en el mundo de los caracteres indeseables.
Así, por ejemplo, si queremos corregir un mensaje de un PRINT debemos oprimir la INST para hacer espacio.
Si, luego de oprimir esta tecla tocamos por error las que mueven el cursor, veremos cómo aparecen signos raros y molestos.

Por ello aquí les dejamos un programa que sale del modo comillas cuando se oprime F1.
El programa se carga en el área asignada al buffer del cassette, por lo que no interfiere con el normal funcionamiento de la computadora.
Se activa a través de SYS850 y se desactiva oprimiendo RUN y RESTORE al mismo tiempo.

Sprites

Aquí les presentamos un pequeño programa que muestra las posibilidades gráficas y animación de sprites multicolor en la Drean Commodore 64.
Este programa imprime el logo de los "Cazafantasmas" y, al terminar, muestra el programa a través de la instrucción LIST.
Nota: ¡shift clr! Se debe oprimir la tecla de SHIFT y CLR al mismo tiempo.
¡com 7!: Se debe oprimir la tecla de Commodore y 7 al mismo tiempo.
¡4 cr ab!: Se debe oprimir la tecla que mueve el cursor hacia abajo cuatro veces.

```
10 REM MODO COMILLAS
20 ML=850
30 FORI=0TO41
40 READA:C=C+A:POKEML+I,A
50 NEXTI
60 IFC<>4484THENPRINT"ERROR EN DATAS. VERIFIQUE LOS
VALORES":STOP
70 PRINT"OPRIMA F1 PARA SALIR DEL MODO COMILLAS"
80 PRINT"SE DESACTIVA CON RUN/STOP"
90 PRINT"SE ACTIVA CON SYS850"
95 SYS850:STOP
100 DATA173,143,2,141,46,3,173,144,2,141
110 DATA47,3,120,169,107,141,143,2,169,3
120 DATA141,144,2,88,96,165,203,201,4,208
130 DATA8,169,0,133,212,133,216,133,199,108,46,3
```

```
10 REM DEMOSTRACION SPRITE MULTICOLOR
20 POKE53281,16:PRINT"!shift clr!!com 7!!!4 cr ab!"
30 FORS=832TO894:READT:POKES,T:NEXT
40 V=53248:POKE2040,13:REM AREA SPRITE
50 POKEV+21,1:REM MUESTRA SPRITE
60 POKEV+28,1:POKEV+37,1:POKEV+38,2:POKEV+39,2:REM ACTIVA
MULTICOLOR
70 POKEV+1,60:REM Y POSICION
80 FORJ=1TO255:POKEV,J:NEXTJ:REM X POSICION
90 LIST
91 DATA0,1,64,0,21,80,0,85
92 DATA64,3,85,64,5,153,240,5
93 DATA85,240,53,85,124,245,153,127
94 DATA65,101,255,213,87,255,197,95
95 DATA195,85,127,65,193,253,87,195
96 DATA253,87,255,213,85,255,85,67
97 DATA253,85,79,81,85,124,15,215
98 DATA240,15,255,240,3,255,192,0
```

GUIA PRACTICA GUIA PRACTICA

LAS SUBRUTINAS DEL DREAN COMMODORE 64

Les explicamos, entre otras, la rutina que barre el teclado y devuelve el código ASCII de la tecla presionada

Nombre de la función: RAMTAS
Propósito: Chequea la RAM
Dirección de llamada: $FF87 (hex) 65415 (dec)
Registros de comunicación: A.X.Y.
Rutina preliminar: Ninguna
Error: Ninguno
Requerimientos de stack: 2
Registros afectados: A.X.Y.
Descripción: Esta rutina cheque a la RAM de la computadora.

Además sirve para determinar el inicio y fin de la RAM.
Otra de las funciones de esta rutina es reajustar los contenidos de las direcciones comprendidas entre $0000 y $0100; $0200 Y $03FF, poner la memoria de pantalla a partir de la dirección $0400 y ajustar el inicio del área correspondiente al buffer del teclado.

Normalmente se llama a esta rutina cuando se inicializa el sistema o cuando se inserta un cartucho.

Pasos a seguir:
1) Acceder a esta rutina
Ejemplo:
a) Desde el Assembler
JSR $FF87
b) Desde el Basic
10 SYS 65415

Nombre de la función: RDTIM
Propósito: Lee el reloj del sistema
Dirección de llamada: $SFFDE (hex) 65502 (dec)
Registros de comunicación: A.X.Y.
Rutina preliminar: Ninguna
Error: Ninguno
Requerimientos de stack: 2
Registros afectados: A.X.Y.
Descripción: Esta rutina se utiliza para leer el reloj del sistema, al que se accede a través de la variable TI. Se actualiza cada 1/60 segundos.
Cuando accedemos a esta rutina, ella devuelve en el acumulador, en el registro X y en el registro Y, loy bytes altos, medios y bajos respectivamente.
Pasos a seguir:
1) Acceder a la rutina.
2) En el Acumulador, X e Y está la hora almacenada en formato byte alto-medio-bajo.
Ejemplos:
a) Desde el Assembler
JSR $ FFDE
STA ALTO: parte alta de la hora
STX MEDIO: parte media.
STY BAJA: parte baja.
b) Desde el Basic
10 SYS 65502: REM ACCEDEMOS A LA RUTINA
20 ALTO=PEEK (780): REM ACUMULADOR EN LA VARIABLE ALTO
30 MEDIO=PEEK (781): REM REG X EN MEDIO
40 BAJO=PEEK (782): REM REG Y EN BAJO
50 END

Nombre de la función: READST
Propósito: Lee la palabra de estado
Dirección de llamada: $FFB7 (hex) 65463 (dec)
Registros de comunicación: A
Rutina preliminar: Ninguna
Error: Ninguno
Requerimientos de stack: 2
Registros afectados: A
Descripción: Esta rutina se utiliza para saber en que estado se encuentran los periféricos de entrada/salida.
Normalmente se utiliza después de una operación que involucre a algunos de los periféricos como lo son Datassette, Drive, impresora, etc.

Una vez que accedemos a la rutina, en el acumulador se devuelve el estado de los periféricos. Cada uno de los 8 bits del acumulador representa un cierto estado.

La tabla 1 los representa.
Pasos a seguir:
1) Acceder a la rutina
2) Decodificar el contenido del acumulador
Ejemplo:
a) Desde el Assembler
JSR $FFB7; accedemos a la rutina
AND H$40; vemos si está en "1"
; el bit 6
BNE EOF; si es así hubo fin de archivo
RTS; terminamos el programa
a) Desde el Basic
10 SYS 65463
20 IFPEEK (780) AND64 = 64 THENPRINT "FIN DE ARCHIVO"
30 END

Nombre de la función: RESTOR
Propósito: Restablece valores utilizados por el sistema operativo.
Dirección de llamada:$FF8A (hex) 65418 (hex).
Registros de comunicación: Ninguno
Rutina preliminar: Ninguna
Error: Ninguno
Requerimientos de stack: 2
Registros afectados:A,X,Y
Descripción: Esta rutina restablece todos los valores (por defecto) utilizados por el Sistema Operativo.
Restablece todos los vectores a sus valores originales.
Pasos a seguir:

Acceder a la rutina

Ejemplo

a) Desde el Assembler:
JSR $FF8A

b) Desde el Basic:

```
10 SYS65418
```

Nombre de la función: SAVE
Propósito: Almacena en disco una sección de memoria
Dirección de llamada: $FFD8 (hex) 65496 (hex).
Registros de comunicación: A,X,Y
Rutina preliminar:SETFLS, SETNAM
Error: 5,8.9. Ver también READST
Requerimientos de stack: Ninguno
Registros afectados: A,X,Y
Descripción: Esta rutina nos permite almacenar una sección de memoria en disco o en cassette. Antes de acceder a ella deberemos, primero, acceder a las rutinas SETNAM y SETFLS las cuales nos permiten determinar el nombre del archivo o programa y los números de dispositivos respectivamente.
Al igual que la rutina LOAD, haremos un ejemplo mucho más completo luego de explicar cómo trabajan las rutinas SETNAM y SETFLS.

Nombre de la función: SCNKEY
Propósito: Barre el teclado devolviendo el código ASCII de la tecla presionada.
Dirección de llamada: $FF9F (hex) 65439 (hex).
Registros de comunicación: Ninguno
Rutina preliminar:IOINIT (opcional).
Error: Ninguno
Requerimientos de stack: 5
Registros afectados: A,X,Y
Descripción: Esta rutina barre el teclado y devuelve en el acumulador el código ASCII de la tecla presionada. Además lo almacena en el buffer de entrada el cual tiene una extensión de 10 bytes (10 caracteres).
Si no se oprimió ninguna tecla, entonces el acumulador tendrá el valor cero (0).
Pasos a seguir:
Acceder a la rutina

Ejemplo

a) Desde el Assembler

```
C000 JSR $FF9F; accedemos a la rutina
COO3 CMP    $00; si no se oprimió ningún caracter
C005 BEQ C000; volvemos a leer otro
C007 JSR $FFD2; si no, lo imprimimos
C00A RTS; finalizamos
```

b) Desde el Basic:

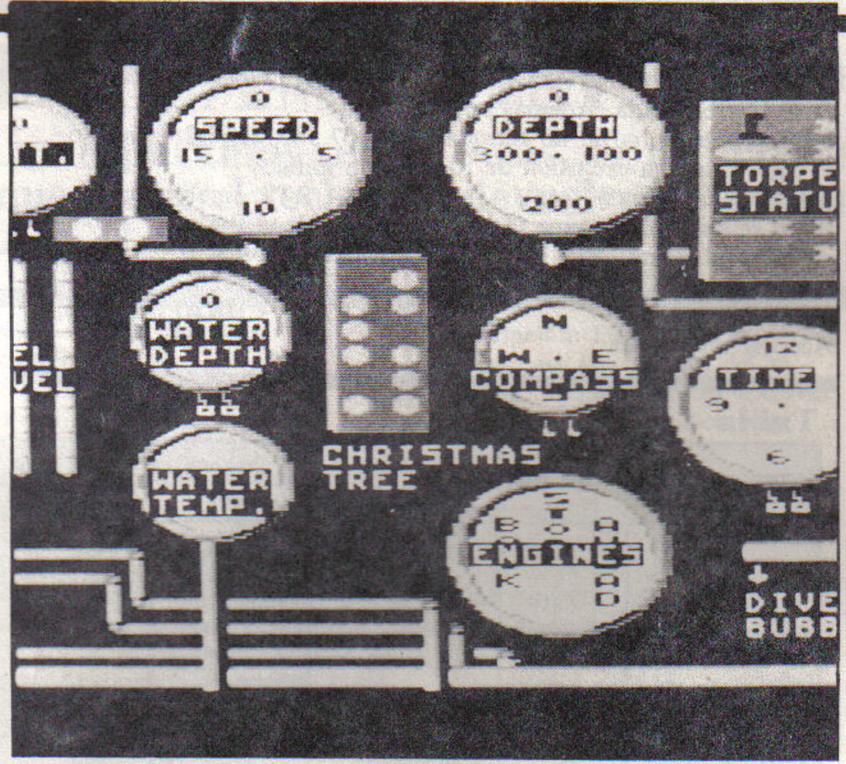
```
10 SYS65439
20 IFPEEK (780)=OTHEN10
20 PRINTCHR$(PEEK(A));
30 END
```

Tabla 1

Posición Bit	Valor Bit	Lectura cassette	L/E Dispositivo serie	Cassette Verif/Load
0	1		Escritura fuera de tiempo	
1	2		Escritura fuera de tiempo	
2	4	Bloque corto		Bloque corto
3	8	Bloque largo		Bloque largo
4	16	Error de lectura irrecuperable		Algún error
5	32	Error de comprobación		Error de comprobación
6	64	Fin de archivo	Linea EOI	
7	128	Fin de cinta	Periférico no presente	Fin de cinta

A: SUPERIOR.	D: REGULAR.
B: MUY BUENO.	E: MALO.
C: BUENO.	F: INACEPTABLE.

SILENT SERVICE



Rating Total: B
Creatividad: B
Profundidad del juego: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 64
Editor: Micro Prose

Nuestro propósito no sólo es comentar las últimas novedades en software para los equipos Drean Commodore sino que, además, nos ocupamos de los que a pesar de su "edad" siguen siendo populares. Silent Service entra dentro de esa categoría. Como simulador de submarinos es el más completo.

En este juego nosotros debemos comandar hábilmente un submarino de la segunda guerra mundial en los mares del Japón.

El programa comienza preguntándonos el escenario de combate. De esta manera podemos participar en distintos tipos de lucha.

En Silent Service es posible practicar tiro, atacar convoyes o luchar contra patrullas enemigas.

Luego de elegir en dónde combatiremos, se nos preguntará sobre las condiciones de lucha. Estas se refieren a la visibilidad (poca o muy buena), a si los barcos enemigos pueden zigzaguear, a la eficiencia de los torpedos (pueden o no detonar), a si vuelven al puerto sólo para reparar los daños, al tipo de destructores del enemigo (duros o "suaves") y a la búsqueda de barcos.

A través del joystick determinamos cada una de las anteriores especificaciones.

El nivel de dificultad se selecciona desde ser un **"midshipman"** (el más fácil) hasta **"Captain"** (el más difícil).

Luego de indicar los datos técnicos, comienza la lucha. El submarino está sobre la superficie trasladándose a una velocidad de 2 nudos.

En el interior se encuentra Juan Cadete quien debe manejarlo. A través del joystick movemos a Juan por el interior de la nave.

Si nos encontramos navegando por la superficie, podemos ir al puente y observar la zona de combate. Claro que, bajo el agua, sólo podemos usar el periscopio para visualizar el escenario, siempre y cuando estemos a la profundidad permitida para accionarlo.

A la derecha tenemos los manómetros y abajo, la bodega. A través de los primeros podemos determinar el estado de los torpedos, la hora del día, la posición de la nave, temperatura de la maquinarias, combustible, niveladores y la carga de las baterías.

Sólo llevamos a Juan hacia abajo cuando necesitamos saber los daños producidos por los destructores enemigos.

Para sumergirnos tenemos que ir a la sala de máquinas y presionar la palanca de mando hacia atrás. Inmediatamente sonará el típico ruido que hacen en las películas cuando un submarino se sumerge y el cartel de desalojar el puente.

A determinada profundidad es factible niverlarlo. Si nos encontramos cerca de la costa deberemos cuidar de no ir demasiado abajo ya que es muy posible chocar contra el fondo.

A través de las teclas de función vamos al puente, vemos nuestra posición en un mapa, observamos los daños del submarino, captamos los barcos enemigos destruidos, subimos el periscopio, nos dirigimos a la sala de máquinas o terminamos el juego.

Si luchamos contra un convoy, debemos cuidar de que los destructores que lo protegen no nos hundan.

Si se nos viene al "humo" tenemos que sumergirnos. Nuestro sonar nos confirmará que se están acercando y, ojalá que no les suceda, que comienzan a tirar cargas de profundidad.

Aquí tenemos que parar las máquinas e ir en completo silencio para no ser localizados por el sonar del destructor. Silent Service es un completísimo simulador de submarinos. Tiene todos los "chiches". Si alguna de las cargas de profundidad nos toca veremos cómo comenzará a hundirse nuestro submarino junto con Juan.

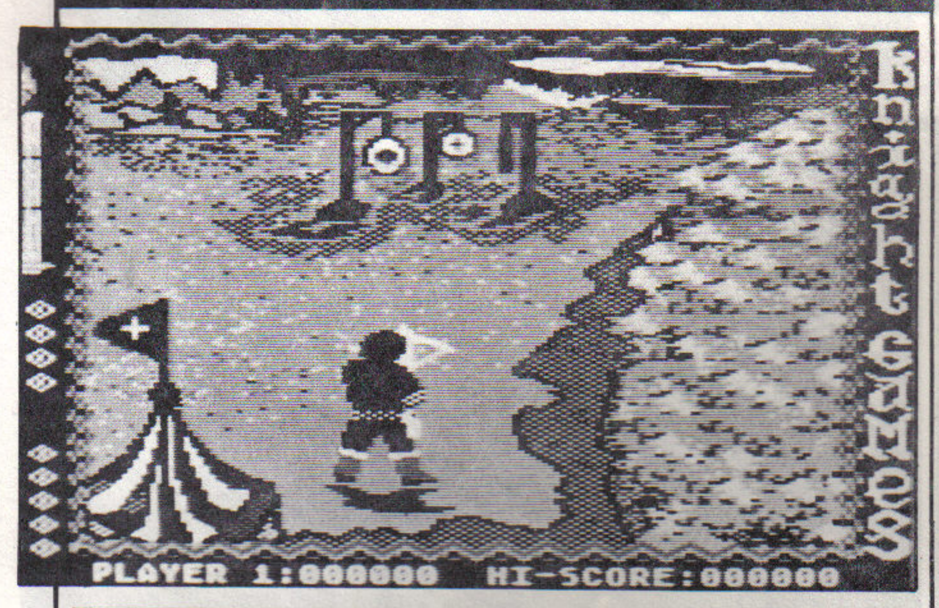
GHOSTS'N GOBLINS

Rating Total: B
Creatividad: B
Documentación: B
Profundidad del juego: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 64
Editor: Elite



Luna llena... Un aullido de lobo se escucha a lo lejos. La fría y espesa niebla imposibilita la visión. El cementerio, con sus lápidas abiertas, nos espera... impaciente".
Sin ir más lejos, éste es el escenario del Ghosts'n Goblins. Nuestro héroe, José Poco Miedo, tendrá que atravesarlo mientras de las tumbas salen muertos vivientes en busca de nuestra vida.
Deberemos cuidarla ya que sólo tenemos cinco.
Junto con el recorrido de José, la música del juego es siniestra y terrorífica, cosa que impide olvidarnos que estamos en un cementerio.
José dispone de lanzas especiales que destruyen instantáneamente a los zoombies.
El cementerio es grande y oscuro, y a medida que se avanza más zoombies aparecen desde las profundidades de la tierra.
En la mitad de la trayectoria tendrá que enfrentar a un cuervo, pero su lanza también puede "despachar" a este pájaro monstruoso.
Desde una pequeña montaña, plantas carnívoras comenzarán a lanzarle "bofes" malditos.
José deberá esquivarlos si no quiere perecer. Si opta por subir, se cuidará mucho de que no le aparezca un muerto de golpe y, por ende, lo mate.
Disparando su lanza, podrá aniquilar a las plantas carnívoras y continuar su recorrido.
Así, deberá enfrentar al pájaro Fénix, gran matador en su época. José tendrá que disparar y pegarle en cuatro oportunidades para que Fénix no moleste más.
Si lo logra, seguirá su camino hacia el lago congelado, donde una isla movible lo llevará hacia la otra orilla.
Si no le pega, no hace falta decir qué le pasará.
Si subimos a la isla y accedemos a la otra orilla, tenemos que esquivar a los pájaros mutantes.
Estos tratarán de agarrarnos y tirarnos al lago congelado, donde se encuentran espantosas bestias.
Si los esquivamos o matamos, podremos continuar la marcha hacia el gran enfrentamiento con el gigante Paragrandote.
En el camino habrá otra planta carnívora que seguirá mandándonos bofes malditos.
Finalmente, los horripilantes pájaros-zapatos, nos mandarán larvas venenosas que deberemos esquivar.
Luego, la gran lucha con Paragrandote. El tratará de, simplemente, aplastarnos con sus grandes pies.
José, con su espada mágica, deberá matarlo para que caiga la llave que nos permitirá pasar al otro nivel.
Ghosts'n Goblins es un simple pero entretenido juego de acción. La música que acompaña a José durante su recorrido es sumamente adecuada para los escenarios que aparecen.
De haber existido este juego en la antigüedad, seguramente el Conde Drácula no hubiese tenido la necesidad de "salir de noche".

KNIGTH GAMES



Rating total: B
Documentación: B
Profundidad del juego: B
Valor en relación al costo: Se justifica
Comp: Drean Commodor 64
Editor: English Software

Hoy, luego de haber visto este juego, hemos comprendido la cantidad "mayúscula" de sangre derramada por los señores feudales y de la "patota" de los Caballeros de la Mesa Redonda. Tal vez sería mejor decir que si en el siglo XV se hubiese inventado la Commodore 64, el escudo, la espada y la armadura se hubiesen cambiado por un joystick.

"Knight Games" es un excelente y fantástico juego que nos lleva rápidamente al siglo XV, durante las competencias que efectuaba la "crem de la crem". Sus Majestades, junto con las doncellas más hermosas de la comarca, asisten a la competencia.

Inicialmente debemos indicarle a la computadora en qué contienda participaremos. Ellas pueden ser Swordfight 1, Swordfight 2, Quarteerstaff, Pikestaff, Archery, Crossbow, Ball & Chain y Axeman. Cada una de ellas está representada por un caballero junto con las armas características de cada combate.

La música —como siempre en los juegos para la Drean Commodore 64— es excelente y en este caso acorde al escenario medieval.

Moviendo el joystick podremos seleccionar el combate. Basta oprimir el botón y luego la tecla 'Y' para confirmar la orden.

La pantalla que sigue a continuación de la anterior establece las opciones del combate. De esta forma acordamos el tiempo de combate (4, 6 u 8 minutos), el tipo de juego (jugador 1 vs computadora o jugador 1 vs jugador 2), si deseamos tener música durante el transcurso del combate y, por último, iniciar el juego.

Si estamos en Swordfight 1 la acción transcurre en la terraza del castillo del Rey. La computadora elige en todos los juegos a Sir Charles Nomeduele para que la represente en los combates.

No hace falta decir que Sir Charles (igualito al de las películas) pega durísimo y, valga la metáfora, "con un fierro". Lady Notehevisto y Lady Nimeacuerdo observan emocionadas el accionar de ambos caballeros (Sir Charles y el que nos representa) quienes golpe a golpe tratarán de ganar el corazón de aquellas hermosuras.

El programa no contempla la posibilidad de querer conquistar, a cambio, el corazón de Negruttso, un esclavo de 2 metros de estatura.

Los indicativos de combate son el puntaje obtenido acorde a los golpes pegados y al tiempo transcurrido. Como en el siglo XV no había cronómetros, el juego utiliza un típico medidor de tiempo: una vela encendida. Cuando ésta se consuma en su totalidad, habrá finalizado la contienda. Debajo de la vela están los contadores parciales los cuales indican los golpes recibidos.

Cuando Sir Chales nos pegó 20 "garrotazos" nuestro caballero caerá irrevocablemente. Además, por cada 10 golpes recibidos, disminuirá el contador principal (presentado a través de escudos).

Los restantes juegos se desarrollan siempre de la misma manera a excepción de Archery y Crossbow (arquería y ballesta) en donde tenemos que pegarle a blancos móviles.

Tenemos que resaltar de "Knight Games" la música y todos los paisajes que se ven durante la competencia, la que trascurre en diversas partes del Reino como ser el bosque real, los valles, el puente elevadizo, salón de fiestas, etc.

Otra característica importante es cómo se han contemplado los movimientos de los caballeros. Se observan todos los detalles.

Por último, nuestro útil consejo: Cuando Sir Charles los vaya a atacar, ustedes díganle 'Sir Charles, gran caballero, tiene usted el cordón del zapato suelto". Cuando ponga cara de estar pensando "¿cómo se me va a desatar el zapato cuando uso armadura especial?" ustedes, ahí mismo, ¡péguenle un garrotazo!!.

STAR PING PONG

Rating Total: B
Creatividad: B
Profundidad del juego: B
Valor en relación al precio: Se justifica
Comp: Drean Commodore 64
Editor: Silvertime

Cuando Hirutsi (ex-campeón mundial de tenis sobre mesa (comunmente llamado ping pong) vio este juego exclamó "Asuky tsi snuo !" dejando completamente sorprendidos a los editores.

Ping Pong es un completísimo simulador de ping pong, que nos permite practicarlo a nivel principiante, medio o superprofesional.

Primeramente debemos seleccionar las condiciones del juego.

Ello posibilitará jugar contra un contrincante o contra la computadora.

También podemos determinar la perspectiva, es decir si veremos el desarrollo del partido desde atrás o desde el costado.

A partir de este primer menú también podemos indicarle a la computadora el modo auto juego, es decir que ella juegue contra si misma.

La velocidad del juego se puede seleccionar entre tres. Así, podemos competir con una velocidad normal, lenta o rápida. Para esta última opción aconsejamos usar un televisor con posibilidad de reproducir en cámara lenta. La pelotita ni se ve!

Completado el primer menú, pasamos al segundo. En éste se selecciona la potencia de cada uno de los golpes posibles de generar. En total son tres, siendo smashing, forehand y backhand.

También es posible modificar el tiempo que nuestra paleta tardará en reaccionar, la velocidad del movimiento y la dureza de nuestro "paletazo".

En el último menú indicamos el color de las paletas (las nuestras y las de las computadora) si cambiamos de lado luego de finalizar un juego y la cantidad de sets que se necesitan para ser los campeones.

Luego de completar los tres menúes, aparece en pantalla la mesa (con red y todo) y las paletas.

Comienza el juego recibiendo la computadora nuestro servicio. El saque es libre, pudiendo dirigir la pelotita hacia cualquier dirección.

El cambio de saque se realiza como en el verdadero ping pong.

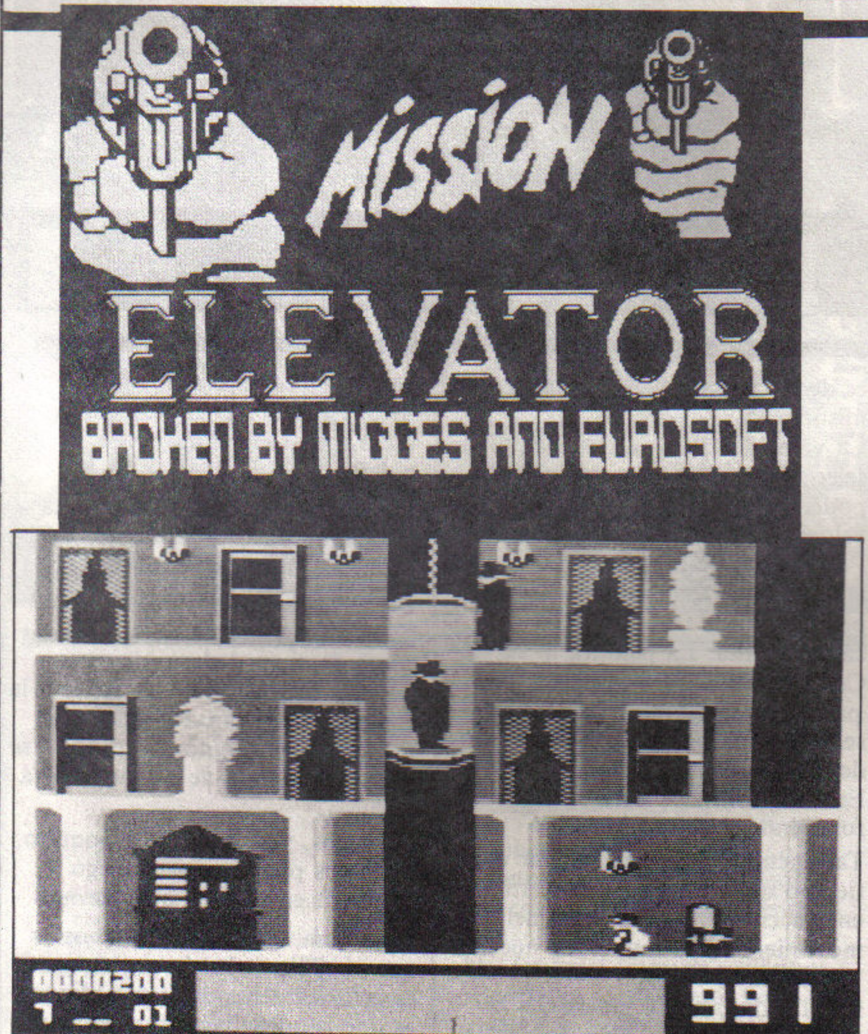
Lo más dirvertido es ver jugar a la computadora contra si misma en el modo más rápido. La pobre "se mata" corriendo de un lado hacia otro.

Nosotros, y siempre de a poquito, podemos practicar para luego competir en el nivel más elevado.

La última noticia que tuvimos es que en China, donde el ping pong es prácticamente deporte nacional, es que las próximas competencias mundiales se harán a través del Ping Pong.

Ojalá que Hirutsi no vuelva a decir esas irreproducibles palabras que tanto mal le hacen al deporte.

MISSION ELEVATOR



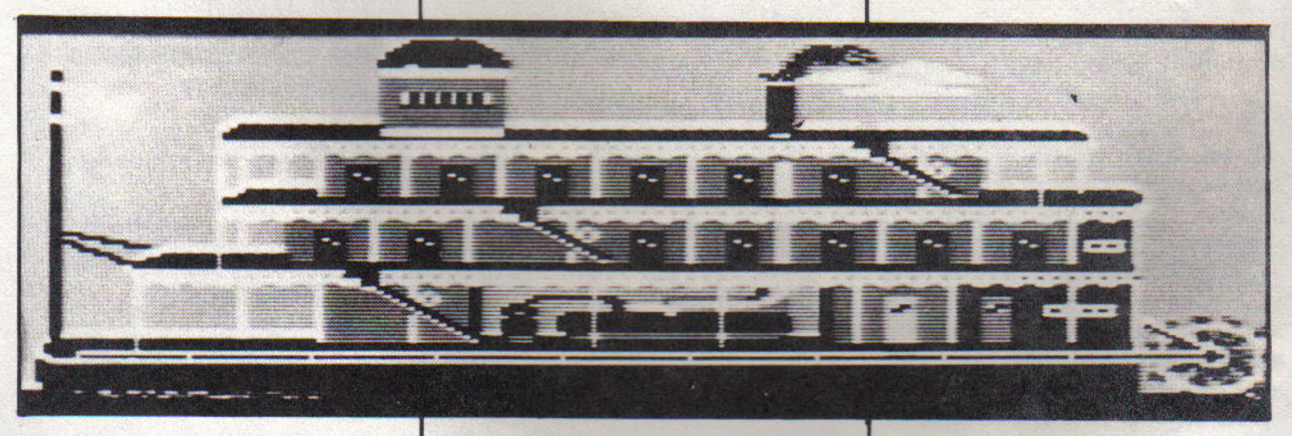
Rating Total: B
Creatividad: B
Documentación: B
Profundidad del juego: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 64
Editor: EUROSOF

A Tony Wait, integrante de la organización mundial de espías y saboteadores, le espera una durísima misión.
Tendrá que ingresar a un edificio enemigo y sabotear sus instalaciones. Mission Elevator comienza mostrándonos una hoja de impresora con información secreta proveniente del cuartel enemigo.
Aquí se nos informa cuál es la situación mientras se pide un "gran valor" para enfrentar a las fuerzas del profesor Andrews Tocoydestruyo.
Luego de ingresar nuestro nombre, comienza la acción. Esta transcurre dentro de la base enemiga que está camuflada como un hotel de varios pisos y muchos ascensores.
Tony tendrá que ir investigando por todo el hotel en busca de claves secretas. Y, al mismo tiempo, esquivar las balas provenientes de revólveres enemigos.
Tony también dispone de un arma. Si el jugador que maneja al protagonista es pacifista y no quiere matar a nadie, puede optar por esquivar las balas, saltando o tirándose cuerpo a tierra.
El hotel está lleno de objetos los cuales Tony deberá investigar en busca de la clave secreta.
Es posible que sobre alguna mesa encuentre dados y éstos lo ayudarán a incrementar su puntaje. La computadora nos pedirá que elijamos un número y que apostemos dinero.
Luego comenzará a tirar los dados. Si sale nuestro número ganaremos dinero, aumentando así nuestro puntaje.
Si perdemos, se reducirá acorde a la suma apostada. El dinero perdido se puede recuperar o apostando de nuevo o, en una de esas, encontrándolo en el piso.
Si nuestro personaje está muy cansado, tiene la posibilidad de sentarse sobre las sillas del hotel para recuperar fuerzas.
Con mucha cautela tendrá que ingresar por alguna de las puertas, siempre y cuando no esté cerrada.
Los secuaces del doctor Andrews son los típicos malos de las películas: sombrero y piloto son sus vestimentas.
Por cada uno que "liquidemos", incrementaremos nuestro puntaje, al igual que si ganamos en los dados.
Tony tiene 8 vidas. Deberemos utilizar un indicador que se encuentra a la izquierda de la pantalla, donde se nos dice las vidas que nos quedan, el piso en que estamos y las claves tomadas.
Moviendo el joystick para abajo podremos analizar qué es lo que hay y, en caso de haber alguna clave, la tomaremos.
Desde ya traten de cuidar que Tony no analice una toma corriente que hay en el segundo piso. Si lo hace, se quedará "hablando pavadas".

MURDER ON THE MISSISSIPPI

Rating Total: B
Creatividad: B
Documentación: B
Profundidad del juego: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 64
Editor: Activision

Seguramente ustedes se habrán dado cuenta que la revisión de software de este número cubrió una buena parte de escenarios históricos.
Competimos en el siglo XV y hundimos barcos japoneses en la Segunda Guerra Mundial. Ahora nos ubicamos a fines del siglo XIX.



"Murder on the Mississippi" (Muerte sobre el Mississippi) es, a diferencia de los anteriores, un juego no violento. Por el contrario, se cambia la tradicional temática guerrera por la de la investigación criminalista.
Al más puro estilo de las novelas de Agatha Cristhie, deberemos descubrir quién fue el que asesinó a un pasajero del barco "Princess of the Mississippi".
Nuestro héroe: Sir Guilliams Quienhasido y su fiel compañero Regis Onono deberán encontrar al culpable de tan horrendo crimen.

Nuestro protagonista, y siempre guiado con el joystick, tendrá que investigar por todos los camarotes del barco e ir interrogando a los pasajeros.
Atrás de él lo acompaña el inseperable Regis quien a través de diálogos le indica si tal puerta está cerrada o si desea ir al puente o a la cubierta del barco.
Cuando Sir Guilliams ingresa a un camarote, puede encontrarse con alguno de los cinco sospechosos. Ellos son un juez, un sacerdote, el maquinista, una hermosa señorita francesa y el capitán del barco.
En total son seis las personas que intervienen. Los anteriores cinco y la víctima, que yace en el piso de su cuarto con la cabeza ensangretada.
Cuando Sir Guilliams ingresa a un camarote, puede encontrarse con alguno de los sospechosos.
De acuerdo con quién se encuentre se ejecutará una música acorde.
Por ejemplo, si ingresamos en la habitación del sacerdote, se entonarán los acordes de una "música gloriosa".
Una vez que estamos dentro podemos comenzar el interrogatorio o analizar evidencias.

Si optamos por lo primera, podemos pedirle al sospechoso que nos hable acerca de la víctima o, si tenemos pruebas suficientes, acusarlo de haber cometido el crimen.
Si tenemos alguna estrategia, podemos pedirle que nos acompañe a algún sitio del barco.
O si no, podemos caminar alrededor del camarote esquivando los objetos que allí se encuentran. Si Sir Guilliams chocase con alguno, por ejemplo con la cama, dirá algo así como "¿qué hace esta cama en mi camino?". Lo mismo ocurre si se estrella contra la puerta o la pared. Para ganar tiempo, lo mejor es ir primero con el capitán y pedirle información acerca de cada uno de los sospechosos.
"Murder on the Mississippi" es un entretenido juego que a más de uno le provocará deseos de meterse por la pantalla y caer sobre el Princess para saber quién fue el asesino de la víctima.
Los diálogos entre Sir Guilliams y Regins son bastantes cómicos. Por ejemplo, cuando ingresan al cuarto de la víctima, el noble le pregunta a Regins "¿qué hace ese hombre durmiendo en piso?" mientras que Regins le contesta "estará descansando".
Nosotros tenemos algunas pruebas. Sospechamos del cadete de la editorial quien al ver este juego comenzó a correr rápidamente hacia la salida del edificio.
¡Ya lo atraparemos!

CORREO-CONSULTAS

BACK-UP

Soy un fiel lector de K64 y Drean Commodore, y poseo una C-64. Unos días atrás, quise pasar unos programas que tenía en cassette a disco, y me encontre con la novedad, de que varios de ellos, al hacer el backup a disco me producían un "OUT OF MEMORY ERROR" en la computadora. Mi pregunta es ¿cómo puedo salvar este inconveniente, y a qué obedece?
También deseo conectarme con usuarios de Commodore 64, e intercambiar conocimientos, programas (tengo unos 600), bibliografía, y todo aquello de importancia para los commodorianos.
Espero que sigan publicando estas dos espléndidas revistas como lo han hecho hasta ahora, son realmente EXCELENTES ya que abarcan distintos niveles de aprendizaje.
Un sincero y cordial saludo.

Fernando M. Deurer
Casa 33, Barrio C.N.E.A.
(8316) Plottier
NEUQUEN

Hay algunos programas copiadores que generan el OUT OF MEMORY cuando se procede a realizar copias de cassette a disco. Por ello te sugerimos que trates de utilizar el más fiable.

PROGRAMAS:

Ante todo reciban mis felicitaciones por su excelente revista, el material siempre es de primer nivel, único en nuestro mercado, sigan adelante y ojalá pronto podamos disfrutar de más páginas para enterarnos de todo lo referido a esta muy buena máquina.
Mi inquietud va a lo siguiente: todos sabemos de la superioridad de un diskette ante un cassette en cuanto a velocidad de carga, rapidez, acopio de datos o sea cuantitativamente, ahora bien, si yo deseo cargar un juego por ejemplo que se caracteriza por su riqueza de imágenes, movimientos, sonido, etc., y poseo un sistema de cassette para efectuarlo, como la fábula del conejo y la tortuga, una vez esta última llega a la meta, ¿cumplen ambos procedimientos (diskette y cassette) idénticos logros en la calidad del programa volcado?
Otra pregunta ¿existe algún procedimiento que me permita ahorrar tiempo cuando cargo con cinta?
Les agradezco por su atención y sigan así con esta buena revista.

José Luis Figueroa. Claypole

El cassette como el diskette cumplen la única función de almacenar información. Claro que uno (el diskette) es mucho más fiable y rápido que el otro.
Es decir que no intervienen en la calidad del programa. El mismo programa almacenado en cinta o en disco corre exactamente igual en ambos casos.
Con respecto a la segunda pregunta la respuesta es sí. El caso más conocido por los usuarios de equipos Drean Commodore es la utilización del Turbo Tape, el cual es un programa que aumenta la velocidad de transferencia entre el datassette y la computadora.

Continuamos con esta sección para que los lectores planteen sus consultas y sugerencias. Para eso deben escribir a Revista para usuarios de Drean Commodore, Paraná 720, 5to. Piso, (1017) Cap.

DREAN COMMODORE 16

Me dirijo a ustedes para preguntarles cuándo van a publicar en su excelente revista los sprites para la Drean Commodore 16, ya que en el manual del usuario no están explicados debidamente.

Marcelo Martínez
La Matanza

La Drean Commodore 16 no tiene posibilidades de manejar sprites. Sin embargo en futuras notas comentaremos cómo realizar programas que simulen la creación de los sprites.

FE DE ERRATAS

En el programa Graficador de Funciones Tridimensional hemos omitido imprimir las siguientes líneas:

```
310 IF EY=0 THEN PRINT"DISTINTA DE CERO":GOTO 300
320 PRINT
330 INPUT"INGRESE EL FACTOR  DE  ESCALA  SOBRE  EL
EJE  !shift Z!";EF
340 IF EF=0 THEN PRINT"DISTINTA DE CERO":GOTO 330
```



Software/Hardware
PEEK
para...
Drean commodore
TENIS INTERNACIONAL
POPEYE
LOGO